

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития  
Кафедра инфокоммуникаций

**ОТЧЕТ**  
**ПО ПРАКТИЧЕСКОЙ РАБОТЕ №11**  
**дисциплины «Программирование на Python»**

Выполнил:

Лейс Алексей Вячеславович  
2 курс, группа ИВТ-б-о-22-1,  
09.03.01 «Информатика и  
вычислительная техника»,  
направленность (профиль)  
«Программное обеспечение средств  
вычислительной техники и  
автоматизированных систем»

---

(подпись)

Руководитель практики: кандидат тех.  
наук доцент кафедры  
инфокоммуникаций: Воронкин Р.А

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2023 г.

**Тема:** Работа с функциями в языке Python

**Цель работы:** приобретение навыков по работе с функциями при написании программ с помощью языка программирования Python версии 3.x.

### Порядок выполнения работы:

Пример 1:

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4
5  import sys
6  from datetime import date
7
8
9  def get_worker():
10     """
11     Запросить данные о работнике.
12     """
13     name = input("Фамилия и инициалы? ")
14     post = input("Должность? ")
15     year = int(input("Год поступления? "))
16
17     # Создать словарь.
18     return {
19         'name': name,
20         'post': post,
21         'year': year,
22     }
23
24
25  def display_workers(staff):
26     """
27     Отобразить список работников.
28     """
29     # Проверить, что список работников не пуст.
30     if staff:
31         # Заголовок таблицы.
32         line = '+--{}+--{}+--{}+--{}+--'.format(
33             '-' * 4,
34             '-' * 30,
35             '-' * 20,
36             '-' * 8
37         )
38         print(line)
39         print(
40             '| {:^4} | {:^30} | {:^20} | {:^8} |'.format(
41                 "No",
42                 "Ф.И.О.",
43                 "Должность",
44                 "Год"
45             )
46         )
47         print(line)
```

```

49         # Вывести данные о всех сотрудниках.
50         for idx, worker in enumerate(staff, 1):
51             print(
52                 '| {:>4} | {:<30} | {:<20} | {:>8} |'.format(
53                     idx,
54                     worker.get('name', ''),
55                     worker.get('post', ''),
56                     worker.get('year', 0)
57                 )
58             )
59             print(line)
60
61     else:
62         print("Список работников пуст.")
63
64
65 def select_workers(staff, period):
66     """
67     Выбрать работников с заданным стажем.
68     """
69     # Получить текущую дату.
70     today = date.today()
71
72     # Сформировать список работников.
73     result = []
74     for employee in staff:
75         if today.year - employee.get('year', today.year) >= period:
76             result.append(employee)
77
78     # Возвратить список выбранных работников.
79     return result
80
81
82 def main():
83     """
84     Главная функция программы.
85     """
86     # Список работников.
87     workers = []
88
89     # Организовать бесконечный цикл запроса команд.
90     while True:
91         # Запросить команду из терминала.
92         command = input(">>> ").lower()
93
94         # Выполнить действие в соответствие с командой.
95         if command == 'exit':
96             break

```

```

97
98     elif command == 'add':
99         # Запросить данные о работнике.
100         worker = get_worker()
101
102         # Добавить словарь в список.
103         workers.append(worker)
104         # Отсортировать список в случае необходимости.
105         if len(workers) > 1:
106             workers.sort(key=lambda item: item.get('name', ''))
107
108     elif command == 'list':
109         # Отобразить всех работников.
110         display_workers(workers)
111
112     elif command.startswith('select '):
113         # Разбить команду на части для выделения стажа.
114         parts = command.split(' ', maxsplit=1)
115         # Получить требуемый стаж.
116         period = int(parts[1])
117
118         # Выбрать работников с заданным стажем.
119         selected = select_workers(workers, period)
120         # Отобразить выбранных работников.
121         display_workers(selected)
122
123     elif command == 'help':
124         # Вывести справку о работе с программой.
125         print("Список команд:\n")
126         print("add - добавить работника;")
127         print("list - вывести список работников;")
128         print("select <стаж> - запросить работников  стажем;")
129         print("help - отобразить справку;")
130         print("exit - завершить работу  программой.")
131
132     else:
133         print(f"Неизвестная команда {command}", file=sys.stderr)
134
135
136 if __name__ == '__main__':
137     main()

```

### Задание:

Решить следующую задачу: основная ветка программы, не считая заголовков функций, состоит из двух строки кода. Это вызов функции `test()` и инструкции `if __name__ == '__main__':`. В ней запрашивается на ввод целое число. Если оно положительное, то вызывается функция `positive()`, тело которой содержит команду вывода на экран слова "Положительное". Если число отрицательное, то вызывается функция `negative()`, ее тело содержит выражение вывода на экран слова "Отрицательное".

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  import sys
4
5  def test():
6      num = (function) def positive() -> None
7      if num > 0:
8          positive()
9      elif num < 0:
10         negative()
11
12     def positive():
13         print("Положительное")
14
15     def negative():
16         print("Отрицательное")
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Windows PowerShell  
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Попробуйте новую кроссплатформенную оболочку PowerShell (<https://aka.ms/pscore6>)

PS C:\Users\LAV\Desktop> & 'D:\LAV\Files\Programs\VisualStudio\Shared\Python39\_64\python.exe' -- 'C:\Users\LAV\Desktop\indiv\_1.py'  
Введите целое число: -7  
Отрицательное  
PS C:\Users\LAV\Desktop> █

В основной ветке программы вызывается функция `cylinder()`, которая вычисляет площадь цилиндра. В теле `cylinder()` определена функция `circle()`, вычисляющая площадь круга по формуле  $S = \pi r^2$ . В теле `cylinder()` у пользователя спрашивается, хочет ли он получить только площадь боковой поверхности цилиндра, которая вычисляется по формуле  $S_{\text{бок}} = 2\pi r h$ , или полную площадь цилиндра. В последнем случае к площади боковой поверхности цилиндра должен добавляться удвоенный результат вычислений функции `circle()`.

```
C:\Users\lav\Desktop> cd indiv_2.py > cylinder

1  import math
2
3  def circle(radius):
4      """
5      Вычисляет площадь круга по формуле:  $\pi * r^2$ .
6      """
7      return math.pi * radius**2
8
9  def cylinder():
10     """
11     Вычисляет площадь цилиндра.
12     Пользователь выбирает, хочет ли он получить только площадь боковой поверхности
13     (S_cylinder_side), или полную площадь цилиндра (S_cylinder_full).
14     """
15     radius = float(input("Введите радиус цилиндра: "))
16     height = float(input("Введите высоту цилиндра: "))
17
18     # Вычисление площади боковой поверхности цилиндра
19     S_cylinder_side = 2 * math.pi * radius * height
20
21     # Вывод результатов площади боковой поверхности
22     print(f"Площадь боковой поверхности цилиндра: {S_cylinder_side}")
23
24     # Пользователь выбирает, хочет ли он получить полную площадь цилиндра
25     full_area_choice = input("Хотите получить полную площадь цилиндра? (yes/no): ").lower()
26
27     if full_area_choice == 'yes':
28         # Вычисление полной площади цилиндра (боковая поверхность + 2 * площадь круга)
29         S_cylinder_full = S_cylinder_side + 2 * circle(radius)
30
31         # Вывод полной площади цилиндра
32         print(f"Полная площадь цилиндра: {S_cylinder_full}")
33
34     if __name__ == "__main__":
35         # Вызов функции cylinder()
36         cylinder()
37
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
launcher '11518' '--' 'C:\Users\lav\Desktop\indiv_2.py'
Введите радиус цилиндра: 45
Введите высоту цилиндра: 70
Площадь боковой поверхности цилиндра: 19792.033717615697
Хотите получить полную площадь цилиндра? (yes/no): yes
Полная площадь цилиндра: 32515.48396465436
PS C:\Users\lav\Desktop>
```

Напишите программу, в которой определены следующие четыре функции:

1. Функция `get_input()` не имеет параметров, запрашивает ввод с клавиатуры и возвращает в основную программу полученную строку.
2. Функция `test_input()` имеет один параметр. В теле она проверяет, можно ли переданное ей значение преобразовать к целому числу. Если можно, возвращает логическое `True`. Если нельзя – `False`.

3. Функция `str_to_int()` имеет один параметр. В теле преобразовывает переданное значение к целочисленному типу. Возвращает полученное число.

4. Функция `print_int()` имеет один параметр. Она выводит переданное значение на экран и ничего не возвращает. В основной ветке программы вызовите первую функцию. То, что она вернула, передайте во вторую функцию.

Если вторая функция вернула `True`, то те же данные (из первой функции) передайте в третью функцию, а возвращенное третьей функцией значение – в четвертую.

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4
5  def get_input():
6      """
7      Запрашивает ввод с клавиатуры и возвращает введенную строку.
8      """
9      user_input = input("Введите значение: ")
10     return user_input
11
12 def test_input(value):
13     """
14     Проверяет, можно ли преобразовать переданное значение к целому числу.
15     Возвращает True, если преобразование возможно, иначе возвращает False.
16     """
17     try:
18         int(value)
19         return True
20     except ValueError:
21         return False
22
23 def str_to_int(value):
24     """
25     Преобразует переданное значение к целочисленному типу и возвращает полученное число.
26     """
27     return int(value)
28
29 def print_int(value):
30     """
31     Выводит переданное значение на экран.
32     """
33     print(value)
34
35 if __name__ == "__main__":
36     # Вызываем первую функцию
37     user_input = get_input()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Windows PowerShell  
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Попробуйте новую кроссплатформенную оболочку PowerShell (<https://aka.ms/pscore6>)

PS C:\Users\LAV\Desktop> & 'D:\LAV\Files\Programs\VisualStudio\Shared\Python39\_64\python.exe' 'c:\Users\LAV\py\launcher' '11937' '--' 'C:\Users\LAV\Desktop\indiv\_2.py'

Введите значение: 17

17

## Индивидуальное задание:

Решить индивидуальное задание лабораторной работы 2.6, оформив каждую команду в виде отдельной функции.

```
4 def add_record(records):
5     """
6     Добавляет запись в список и сортирует erd по расчетному счету плательщика.
7     """
8     record = {}
9     record['расчетный счет плательщика'] = input('Введите расчетный счет плательщика: ')
10    record['расчетный счет получателя'] = input('Введите расчетный счет получателя: ')
11    record['перечисляемая сумма в руб.'] = float(input('Введите перечисляемую сумму в рублях: '))
12    records.append(record)
13    records.sort(key=lambda x: x['расчетный счет плательщика'])
14    print('Запись добавлена и отсортирована.')
15
16 def display_records(records):
17     """
18     Выводит таблицу g данными.
19     """
20    line = '+-{}-+-{}-+-{}-+'.format('-'*20, '-'*20, '-'*25)
21    print(line)
22    print(' | {:^20} | {:^20} | {:^25} | '.format("g. счет плательщика", "g. счет получателя", "Перечисляемая сумма"))
23    print(line)
24
25    for record in records:
26        print(' | {:<20} | {:<20} | {:>25} | '.format(record['расчетный счет плательщика'],
27                                                       record['расчетный счет получателя'],
28                                                       record['перечисляемая сумма в руб.']))
29    print(line)
30
31 def select_record(records):
32     """
33     Поиск суммы по расчетному счету плательщика.
34     """
35    rc = input('Введите расчетный счет плательщика для поиска суммы: ')
36    found = False
37
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Windows PowerShell  
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Попробуйте новую кроссплатформенную оболочку PowerShell (<https://aka.ms/pscore6>)

PS C:\Users\LAV\Desktop> & 'D:\LAV\Files\Programs\VisualStudio\Shared\Python39\_64\python.exe' 'c:\Users\LAV\.vscode\extensions\ms-python.pyth  
'--' 'C:\Users\LAV\Desktop\indiv.py'  
Введите команду (add, display, select, exit):

## Ответы на вопросы:

1. Каково назначение функций в языке программирования Python?
  - Функции в Python используются для группировки кода внутри блока, который можно вызывать многократно. Они облегчают повторное использование кода и делают программу более структурированной.
2. Каково назначение операторов `def` и `return`?
  - Оператор `def` используется для определения функций, а оператор `return` - для возвращения значения из функции. `def` задает блок кода, а `return` указывает, что функция должна вернуть определенное значение.



3. Каково назначение локальных и глобальных переменных при написании функций в Python?

- Локальные переменные существуют только внутри функции и не видны извне. Глобальные переменные объявляются вне функций и могут быть видны в любой части программы.

4. Как вернуть несколько значений из функции Python?

- Несколько значений можно вернуть, разделив их запятой: ``return value1, value2``. Эти значения будут упакованы в кортеж.

5. Какие существуют способы передачи значений в функцию?

- Значения могут передаваться в функцию позиционно, по имени (ключевым словом) или через аргументы со значениями по умолчанию.

6. Как задать значение аргументов функции по умолчанию?

- Значения аргументов по умолчанию задаются в определении функции, например: ``def my_function(arg1, arg2=default_value)``.

7. Каково назначение lambda-выражений в языке Python?

- Lambda-выражения создают анонимные функции. Они полезны для создания небольших функций, которые используются только в конкретном контексте.

8. Как осуществляется документирование кода согласно PEP257?

- Документация кода в Python обычно осуществляется с использованием строк документации (docstrings). PEP257 определяет стандарты для написания этих строк, включая форматирование и структуру.

9. В чем особенность однострочных и многострочных форм строк документации?

- Однострочные строки документации обычно используются для краткого описания функции, класса или модуля. Многострочные строки документации обеспечивают более подробное описание и могут охватывать несколько строк кода.

Вывод В ходе работы приобрёл навыки по работе с функциями при написании программ с помощью языка программирования Python версии 3.x.