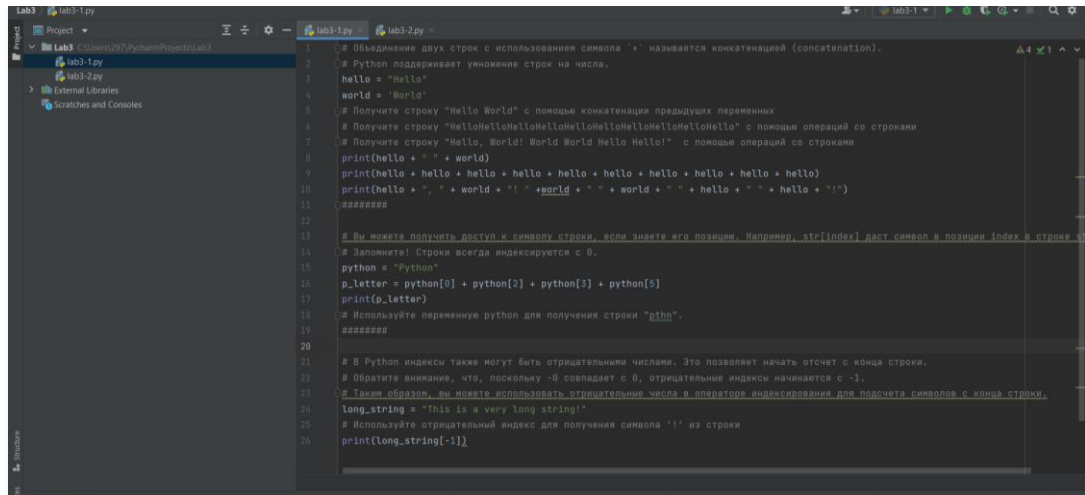


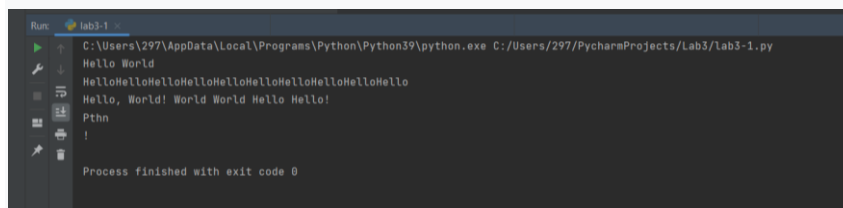
# Отчёт по практической работе № 3

(Выполнил: студент КЭ-103 Евстигнеев Дмитрий)

1. Создан файл «lab3-1.py». Скопирован текст из текстового файла «3.1.txt». Прочитаны комментарии в тексте программы. Выполнены задания в комментариях.

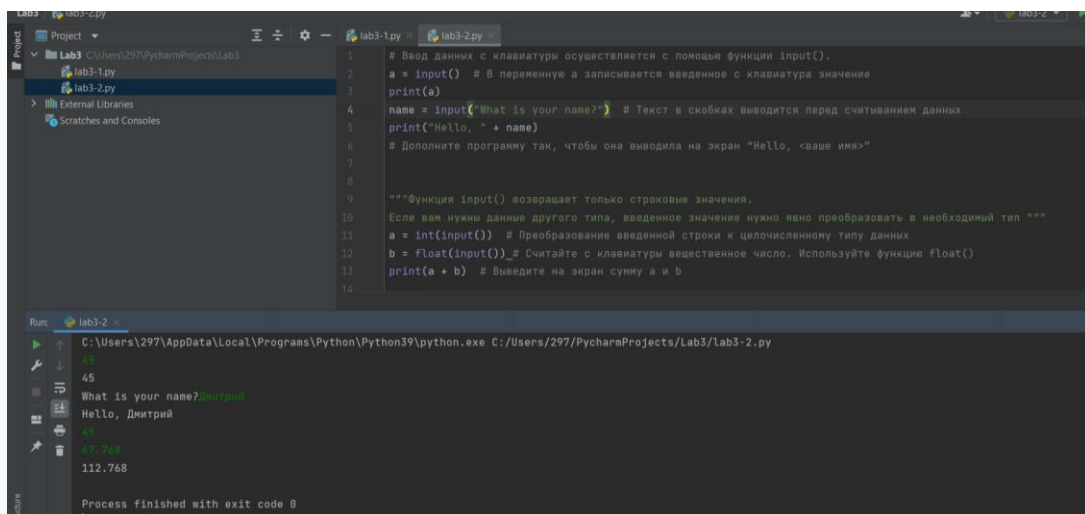


```
1 # Объединение двух строк с использованием символа '+' называется конкатенацией (concatenation).
2 # Python поддерживает умножение строк на числа.
3 hello = "Hello"
4 world = "World"
5 # Получите строку "Hello World" с помощью конкатенации предыдущих переменных
6 # Получите строку "HelloHelloHelloHelloHelloHelloHelloHello" с помощью операций со строками
7 # Получите строку "Hello, World! World World Hello Hello!" с помощью операций со строками
8 print(hello + " " + world)
9 print(hello + hello + hello + hello + hello + hello + hello + hello + hello)
10 print(hello + ", " + world + "! " + world + " " + world + " " + hello + " " + hello + "!")
11 #####
12 # Вы можете получить доступ к символу строки, если знаете его позицию. Например, str[index] даст символ в позиции index в строке str.
13 # Помните! Строки всегда индексируются с 0.
14 python = "python"
15 p_letter = python[0] + python[2] + python[3] + python[5]
16 print(p_letter)
17 # Используйте переменную python для получения строки "pthn".
18 #####
19 # В Python индексы также могут быть отрицательными числами. Это позволяет начать отсчет с конца строки.
20 # Обратите внимание, что, поскольку -0 совпадает с 0, отрицательные индексы начинаются с -1.
21 # Таким образом, вы можете использовать отрицательные числа в операторе индексирования для подбора символов с конца строки.
22 long_string = "This is a very long string!"
23 # Используйте отрицательный индекс для получения символа '!' из строки
24 print(long_string[-1])
```



```
Run: lab3-1
C:\Users\297\AppData\Local\Programs\Python\Python39\python.exe C:\Users\297\PycharmProjects\Lab3\Lab3-1.py
Hello World
HelloHelloHelloHelloHelloHelloHelloHelloHello
Hello, World! World World Hello Hello!
pthn
!
Process finished with exit code 0
```

2. Создан файл «lab3-2.py». Скопирован текст из текстового файла «3.2.txt». Прочитаны комментарии в тексте программы. Выполнены задания в комментариях.



```
1 # Ввод данных с клавиатуры осуществляется с помощью функции input().
2 a = input() # В переменную a записывается введенное с клавиатуры значение
3 print(a)
4 name = input("What is your name?") # Текст в скобках выводится перед считыванием данных
5 print("Hello, " + name)
6 # Дополните программу так, чтобы она выводила на экран "Hello, <ваше имя>"
7
8
9 """Функция input() возвращает только строковые значения.
10 Если вам нужны данные другого типа, введенное значение нужно явно преобразовать в необходимый тип """
11 a = int(input()) # Преобразование введенной строки к целочисленному типу данных
12 b = float(input()) # Считайте с клавиатуры вещественное число. Используйте функции float()
13 print(a + b) # Выведите на экран сумму a и b
14
```

3. Создан файл «lab3-3.py». Скопирован текст из текстового файла «3.2.txt». Прочитаны комментарии в тексте программы. Выполнены задания в комментариях.

```

4  Например, str[2:5]
5
6  """
7  monty_python = "Monty Python"
8  print(monty_python[1:5])
9  monty = monty_python[:5]  # Один или оба индекса могут быть отброшены.
10 monty_python[:5]  # Эквивалентен выражению monty_python[0:5]
11 print(monty)
12 python = monty_python[6:]  # Используйте оператор среза для получения подстроки "Python"
13 print(python)
14
15 # Используйте ключевое слово in для проверки того, содержится ли подстрока в строке.
16 ice_cream = "ice cream"
17 print("cream" in ice_cream)  # Обратите внимание на результат операции
18 # Проверьте есть ли подстроки "ice", "Ice" и "ice " в строке "ice cream". Выведите результаты каждой из проверок.
19 print("ice" in ice_cream)
20 print("Ice" in ice_cream)
21 print("ice " in ice_cream)
22
23 # Функция len() используется для подсчета количества символов в строке
24 # Символы """ и """ используются для создания многострочной строки
25 phrase = """
26 It is a really long string
27 triple-quoted strings are used
28 to define multi-line strings
29 """
30 # Получите первую половину строки, хранящейся в переменной phrase.
31 # Примечание: не забудьте о преобразовании типов.
32 print(len(phrase))
33 first = phrase[0:45]

```

```

Run: lab3-3
C:\Users\297\AppData\Local\Programs\Python\Python39\python.exe C:/Users/297/PycharmProjects/Lab3/Lab3-3.py
onty
Monty
Python
True
True
False
True
88

It is a really long string
triple-quoted str

Process finished with exit code 0

```

- Создан файл «lab3-4.py». Скопирован текст из текстового файла «3.4.txt». Прочитаны комментарии в тексте программы. Выполнены задания в комментариях.

```

6  """
7  dont_worry = "Don't worry about apostrophes"
8  print(dont_worry)
9  print("\Sweet" is an ice-cream")
10 print('text')
11 # Выведите в консоль следующий текст с помощью одной строки: The name of this ice-cream is "Sweet'n'Tasty"
12 print("The name of this ice-cream is \Sweet'n'Tasty")
13
14 # Существует множество полезных методов для работы со строками.
15 # Вы можете использовать функцию lower() для преобразования всех прописных символов в строке в строчные
16 # Функция upper() используется для перевода всех строчных символов в строке в прописные
17 # Чтобы использовать любой строковый метод, поставьте точку в конце строки (или переменной, содержащей строку) и введите название
18 # В PyCharm Вы можете изучить все доступные строковые методы используя сочетание клавиш Ctrl+Space после точки
19
20 monty_python = "Monty Python"
21 print(monty_python)
22 print(monty_python.lower())
23 print(monty_python.upper())
24
25 # Обратите внимание на результат применения метода lower()
26 # Выведите в консоль строку в верхнем регистре
27
28 # Оператор %, напечатанный после строки, используется для объединения строки с переменными.
29 # Оператор % заменит %s в строке строковой переменной, идущей за оператором %.
30 # Специальный символ %d используется в качестве зарезервированного места в строке для переменной, имеющей числовое или десятичное
31
32 name = "John"
33 print("Hello, PyCharm! My name is %s!" % name) # Обратите внимание, что %s расположено внутри строки, а символ % расположен после строки
34 years = 18
35 print("I'm %d years old" % years) # Выведите в консоль свой возраст

```

```

Run: lab3-4
C:\Users\297\AppData\Local\Programs\Python\Python39\python.exe C:/Users/297/PycharmProjects/Lab3/Lab3-4.py
Don't worry about apostrophes
Sweet is an ice-cream
text
The name of this ice-cream is Sweet'n'Tasty
Monty Python
monty python
MONTY PYTHON
Hello, PyCharm! My name is John!
I'm 18 years old

Process finished with exit code 0

```

5. Создан файл «lab3-5.py». Написана программа, которая запрашивает и считывает с клавиатуры мои данные: фамилия, имя, группа. Далее программа выдаёт текст:
- «Привет, <фамилия> <имя> из группы <группа>!  
Введи свою электронную почту?»**

Далее программа ожидает ввода почты и возвращает на экран зашифрованный следующим образом текст: 5 символов фамилии + 5 символов имени (дважды) + 5 символов электронной почты (трижды), все символы в нижнем регистре.

```
lab3-5.py
1 name = input("Введите имя") #Дмитрий
2 surname = input("Введите фамилию") #Евстигнеев
3 group = input("Введите номер группы") #КЭ-103
4
5 print("Привет, %s % surname + " %s " % name + "из группы %s! \nВведи свою электронную почту?" % group)
6 mail = input("Ввод почты") #yevstigneev34@mail.ru
7
8 en_text__ = surname[:5] + 2*(name[:5]) + 3*(mail[8:13])
9 #Имя и фамилия с 0 по 4 символ включительно, почта с 8 по 12 символ включительно
10 print("Зашифрованный текст:", en_text.lower())
11
```

```
Run: lab3-5
C:\Users\297\AppData\Local\Programs\Python\Python39\python.exe C:/Users/297/PycharmProjects/Lab3/lab3-5.py
Введите имяДмитрий
Введите фамилиюЕвстигнеев
Введите номер группыКЭ-103
Привет, Евстигнеев Дмитрий из группы КЭ-103!
Введи свою электронную почту?
Ввод почтыyevstigneev34@mail.ru
Зашифрованный текст: евстидмитрмдмитреев34еев34еев34

Process finished with exit code 0
```

6. Создан файл «lab3-6.py». Реализована программа, которая считывает целое число  $1 < N < 10$  с клавиатуры и печатает на экран N изображений в одну строку

```
Lab3 - lab3-6.py
1 pic1 = "  _ _  "
2 pic2 = " (o o) "
3 pic3 = " / V \ "
4 pic4 = " / ( _ ) \ "
5 pic5 = " AA AA "
6 N = int(input("Введите число N в интервале (1,10)"))
7 print(pic1 * N)
8 print(pic2 * N)
9 print(pic3 * N)
10 print(pic4 * N)
11 print(pic5 * N)
```

```
Run: lab3-6
C:\Users\297\AppData\Local\Programs\Python\Python39\python.exe C:/Users/297/PycharmProjects/Lab3/lab3-6.py
Введите число N в интервале (1,10)5
  _ _  (o o) / V \ / ( _ ) \ AA AA
  _ _  (o o) / V \ / ( _ ) \ AA AA
  _ _  (o o) / V \ / ( _ ) \ AA AA
  _ _  (o o) / V \ / ( _ ) \ AA AA
  _ _  (o o) / V \ / ( _ ) \ AA AA

Process finished with exit code 0
```