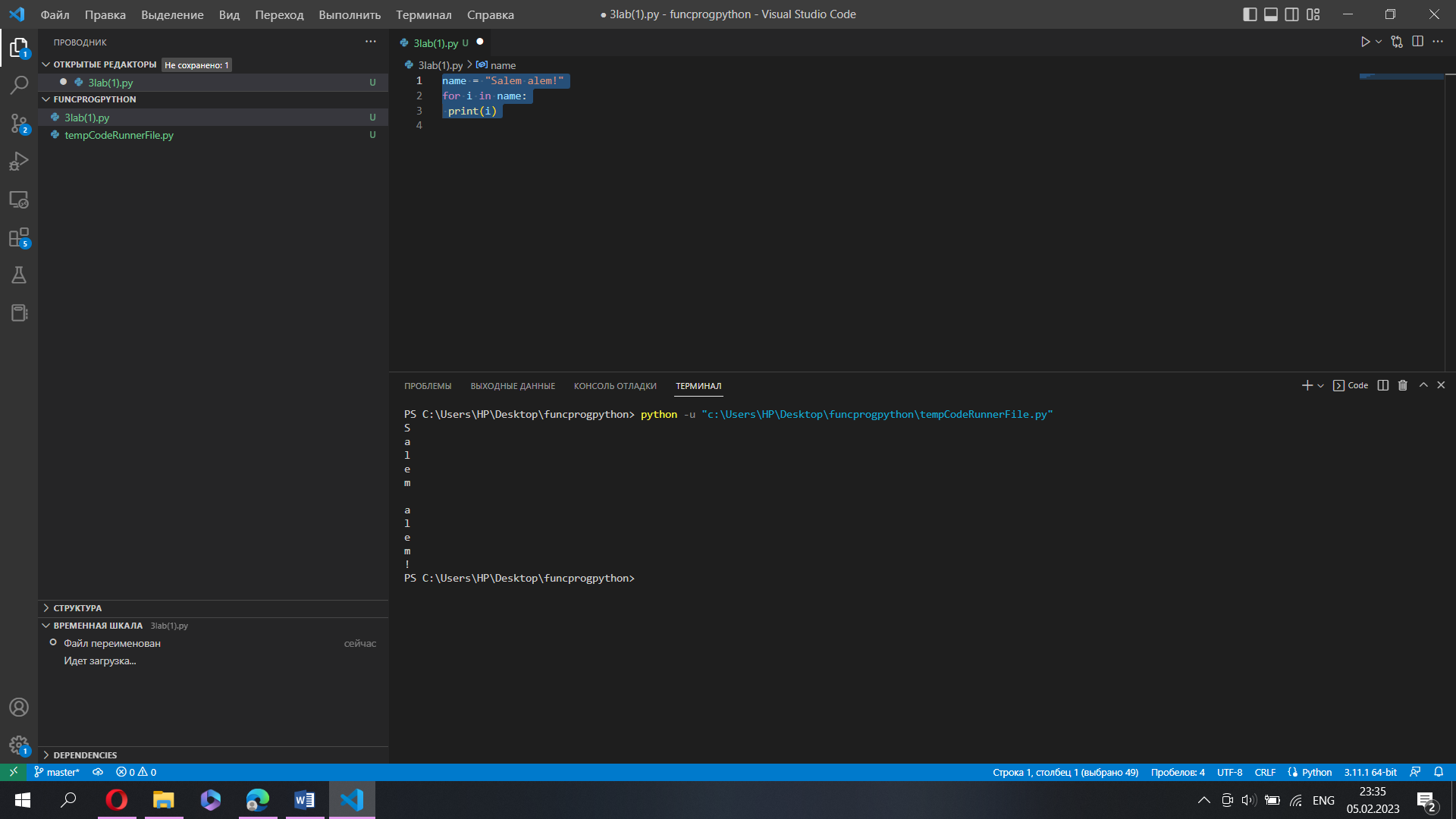
Лабораторная работа №3

#1) Написать программу используя цикл for и while.

name = "Salem alem!"

for i in name:

 print(i)

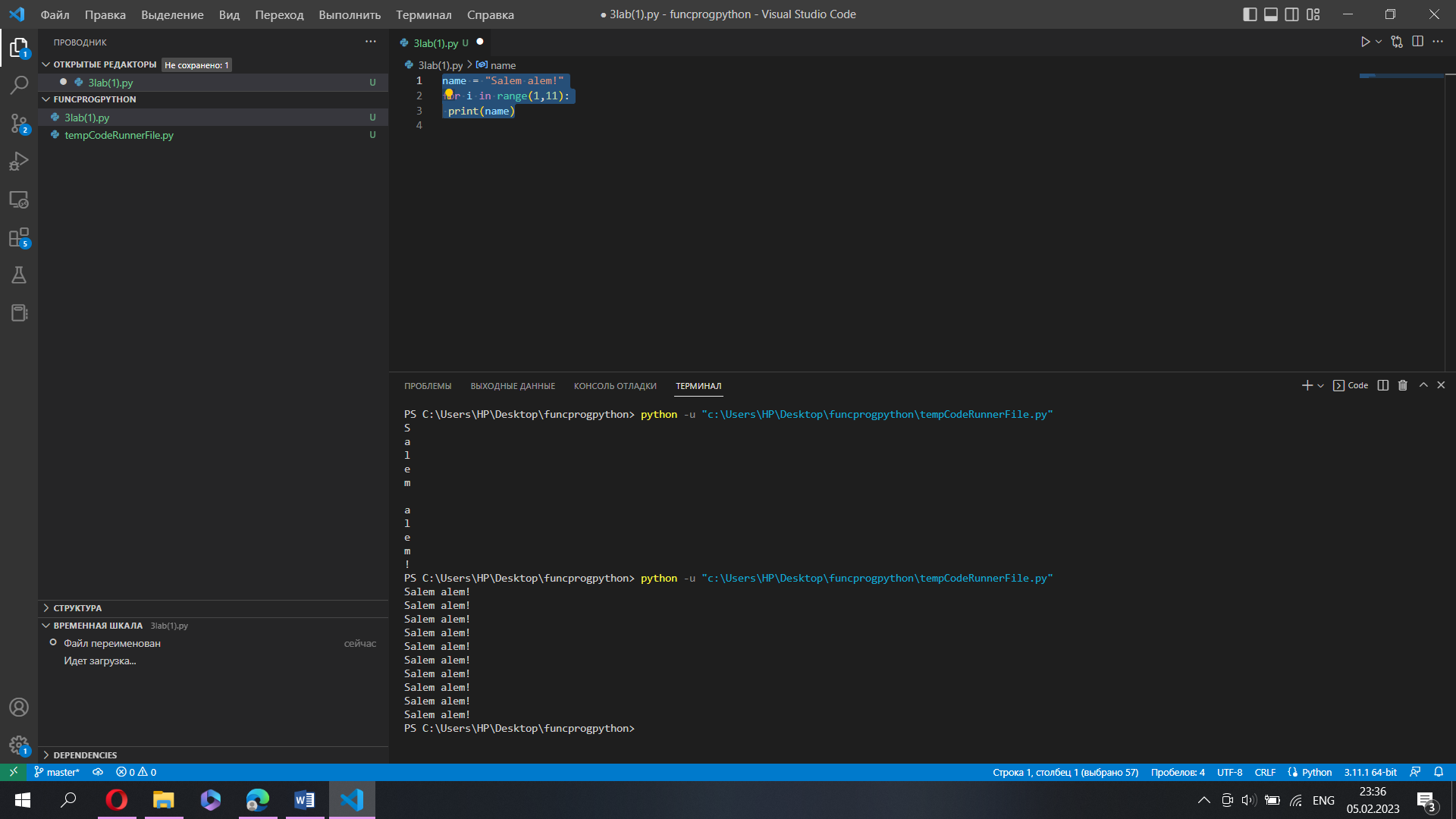


#Например я хочу вывести нашу фразу 5 раз, что нам необходимо сделать?

name = "Salem alem!"

for i in range(1,11):

 print(name)



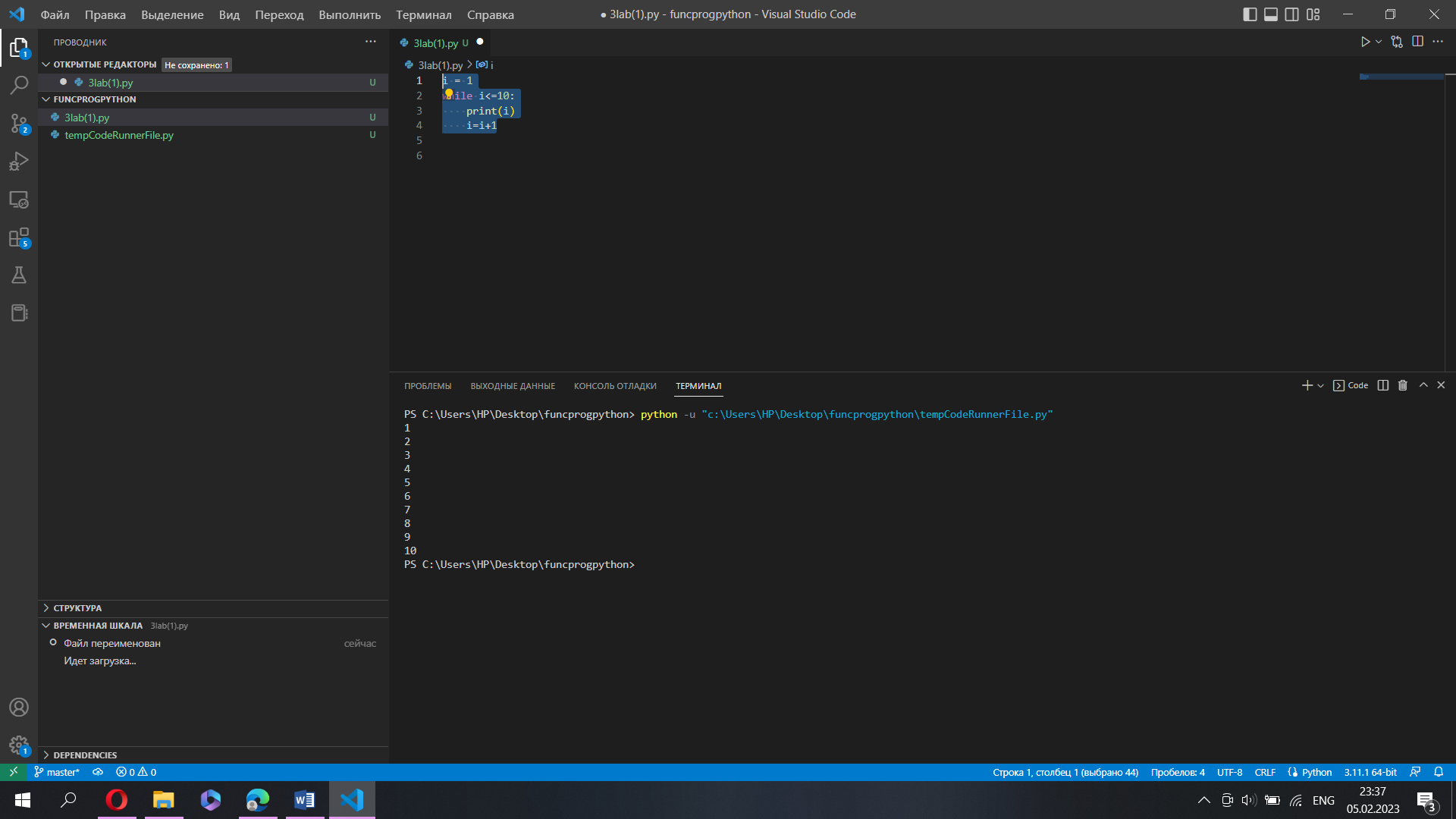
#цикл while

i = 1

while i<=10:

    print(i)

    i=i+1



#цикл while

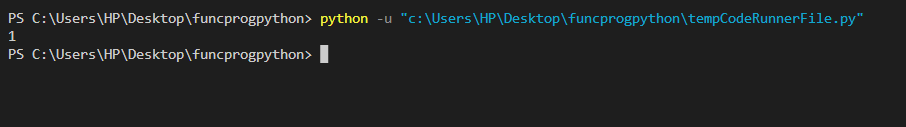
i = 1

while i<=10:

    print(i)

    i=i+1

    break



#цикл while

i = 1

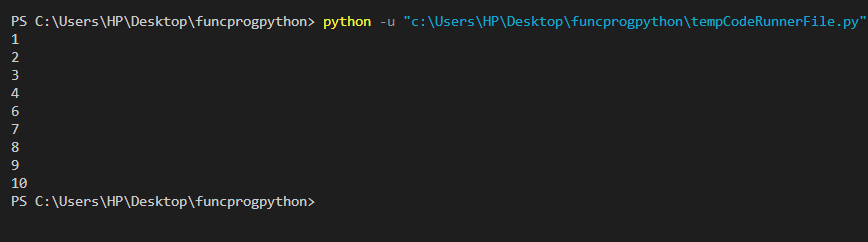
while i<=10:

    if i!=5:

     print(i)

    i=i+1

    continue

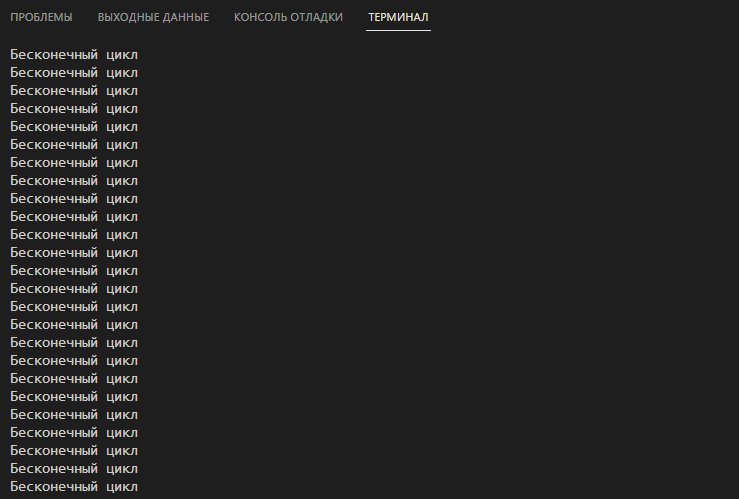


#цикл while

i = 1

while 1==1:

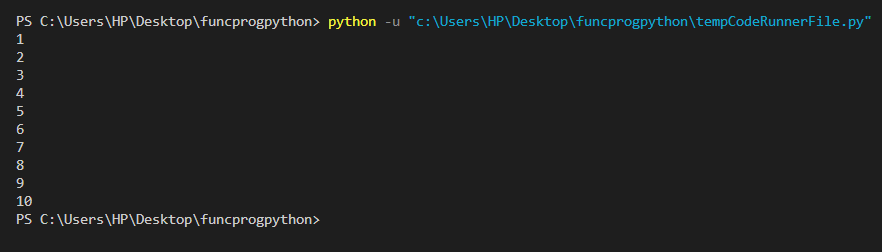
    print("Бесконечный цикл")



#2) используя функцию range() сделать список. в функцию range() введите данные с разными типами и выведите на экран в разных примерах.

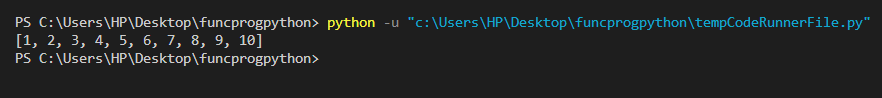
for number in range(1,11):#первое число это начинающее число, последняя заканчивающаю

  print(number)#первая и последняя включена



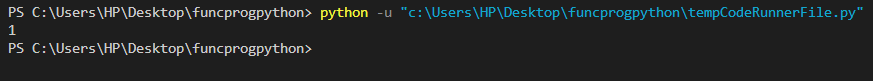
x=list(range (1,11))

print(x)



for number in range(1,11,13):

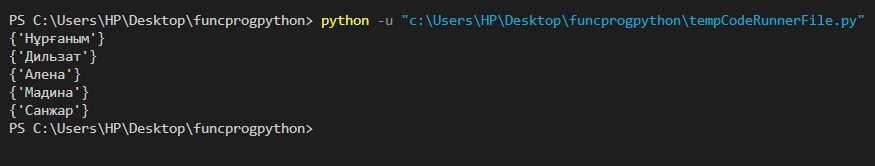
    print(number)



friends\_list = ["Нурали", "Нұрғаным","Қасым", "Дильзат", "Зейнеп", "Алена", "Аймұрат", "Мадина", "Алихан", "Санжар"]

for invite in range(1, len(friends\_list),2 ):

     print({friends\_list[invite].title()} )



#3) используйте функции randint() randrange() random() enumerate()  в своей программе

print("Выбирите число, и получите в подарок упаковку чая!")

from random import randrange

tea\_list = ['ЧЕРНЫЙ ЧАЙ', 'ЗЕЛЕНЫЙ ЧАЙ', 'БЕЛЫЙ ЧАЙ', 'УЛУН', 'ЧАЙ МАТЕ']

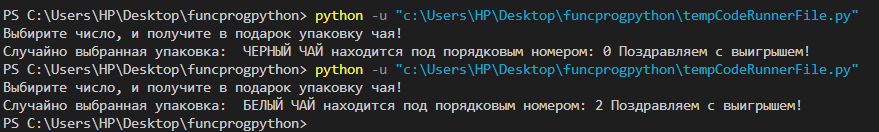
# получаем случайный номер индекса

i = randrange(len(tea\_list))

item = tea\_list[i]

# Выбераем элемент, используя порядковый номер

print("Случайно выбранная упаковка: ", tea\_list[i], "находится под порядковым номером:", i, "Поздравляем с выигрышем!")

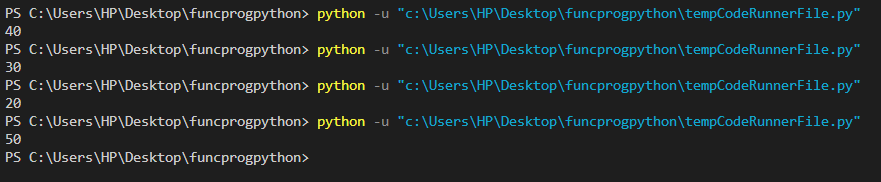


import random

number\_list = [10, 20, 30, 40, 50]

# случайный элемент из списка

print(random.choice(number\_list))



korzina = ['нан', 'май', 'сүт']

enumerateKorzina = enumerate(korzina)

print(type(enumerateKorzina))

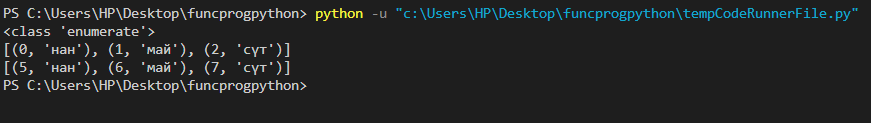
# собираем в список

print(list(enumerateKorzina))

# изменяем

enumerateKorzina = enumerate(korzina, 5)

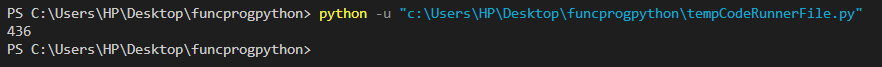
print(list(enumerateKorzina))



import random

number = random.randint(100, 1000)

print(number)



4) решите следующие задачи:

1. Даны два целых числа A и B (при этом A ≤ B). Выведите все числа от A до B включительно.

a = int(input())

b = int(input())

if a>b:

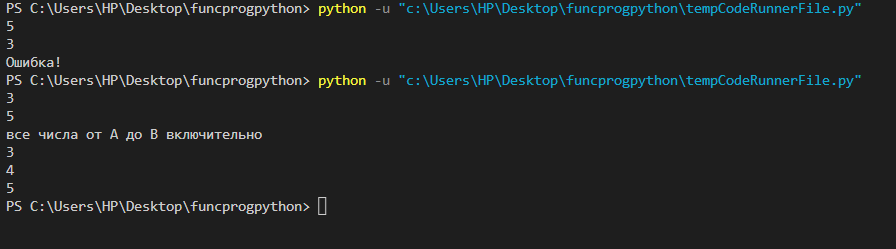
 print("Ошибка!")

else:

 print("все числа от A до B включительно")

for i in range(a, b + 1):

    print(i)



2. Даны два целых числа A и В. Выведите все числа от A до B включительно, в порядке возрастания, если A < B, или в порядке убывания в противном случае.

a=int(input())

b=int(input())

if a<b:

    print("a<b выводим по возрастанию")

    for i in range(a, b+1):

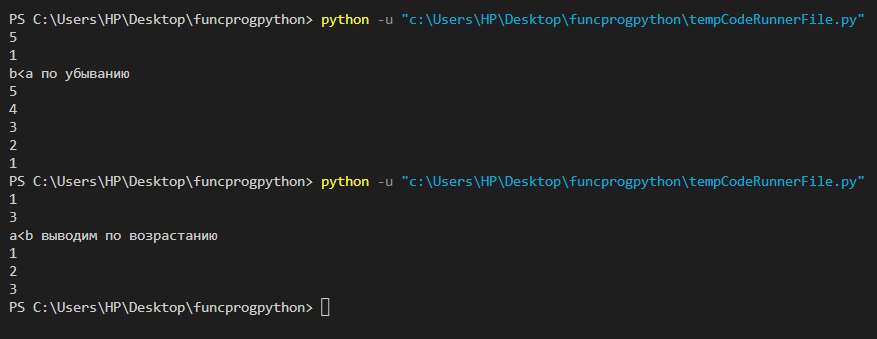
        print(i)

else:

    print("b<a по убыванию")

    for i in reversed(range(b, a+1)):

        print(i)



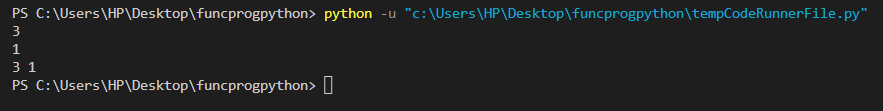
3. Даны два целых числа A и В, A>B. Выведите все нечётные числа от A до B включительно, в порядке убывания. В этой задаче можно обойтись без инструкции if.

a = int(input())

b = int(input())

for i in range(a - (a + 1) % 2, b - b % 2, -2):

    print(i, end=' ')



4. Для настольной игры используются карточки с номерами от 1 до N. Одна карточка потерялась. Найдите ее, зная номера оставшихся карточек.

Дано число N, далее N − 1 номер оставшихся карточек (различные числа от 1 до N). Программа должна вывести номер потерянной карточки.

Для самых умных: массивами и аналогичными структурами данных пользоваться нельзя.

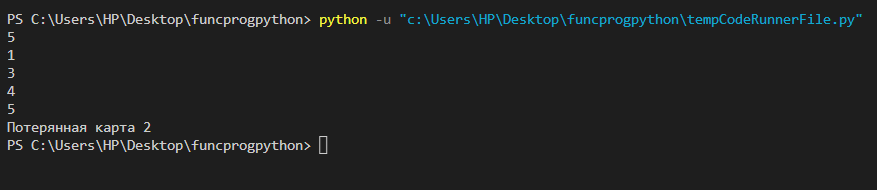
n = int(input())

s = 0

for i in range(n-1):

 s += int(input())

print("Потерянная карта {}".format((n+1)\*n//2 - s))



range\_list = int(input("Сколько карточек у нас?   "))

num\_remove = int(input("Какую карточку удалить?   "))

a = [int for int in range(1,range\_list+1)]

a.pop(num\_remove-1)

print(a)

x = sum(range(1,range\_list+1))

y = sum(a)

z = x - y

print('Сумма диапазона: %d \nСумма значений: %d \nОтсутствует значение:  %d' % (x, y, z))

