

Задание 1.

Исходные данные:

Реализовать функцию, принимающую два числа (позиционные аргументы) и выполняющую их деление. Числа запрашивать у пользователя, предусмотреть обработку ситуации деления на ноль.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> a = 7.5
>>> b = 2.5
>>> def my_del(a, b):
...     try:
...         a = float(input("Введите целое число: "))
...         b = float(input("Введите целое число: "))
...     except zerodivisionerror:
...         return
...     s = a / b
...     return s
...
>>> print(my_del(a, b))
Введите целое число: 25
Введите целое число: 5
5.0
>>>
```

Задание 2.

Исходные данные:

Выполнить функцию, которая принимает несколько параметров, описывающих данные пользователя: имя, фамилия, год рождения, город проживания, email, телефон. Функция должна принимать параметры как именованные аргументы. Осуществить вывод данных о пользователе одной строкой.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> a = 'Name'
>>> b = 'Family name'
>>> c = 'Year of birth'
>>> d = 'Current city'
>>> e = 'Email'
>>> f = 'Phone'
>>> my_list_1 = [a, b, c, d, e, f]
>>> def my_list_1(a, b, c, d, e, f):
...     try:
...         a = input("Введите Ваше имя: ")
...         b = input("Введите Вашу фамилию: ")
...         c = input("Введите Ваш год рождения: ")
...         d = input("Введите Ваш город проживания: ")
...         e = input("Введите Ваш эмайл: ")
```

```

...     f = input("Введите Ваш телефон: ")
...     except ValueError:
...         return
...     my_list_1 = ",".join(my_list_1)
...     return my_list_1
...
>>> print(my_list_1(a, b, c, d, e, f))
Введите Ваше имя: Def
Введите Вашу фамилию: Felloy
Введите Ваш год рождения: one thousand nine hundred twenty
Введите Ваш город проживания: Landan
Введите Ваш email: def@gmail.com
Введите Ваш телефон: xxxxxxxxxxxx
'Def,Felloy,one thousand nine hundred twenty,Landan,def@gmail.com,xxxxxxxxxxx'>>

```

Задание 3.

Исходные данные:

Реализовать функцию `my_func()`, которая принимает три позиционных аргумента и возвращает сумму наибольших двух аргументов.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> print(my_func(a, b, c))
```

```
>>> a = [2, 5, 7]
```

```
>>> b = [9, 12, 15]
```

```
>>> c = [18, 21, 24]
```

```
>>> def my_func(a, b, c):
```

```
...     a = [2, 5, 7]
```

```
...     b = [9, 12, 15]
```

```
...     c = [18, 21, 24]
```

```
...     d = max(b, c)
```

```
...     return d
```

```
...
```

```
>>> print(my_func(a, b, c))
```

```
[18, 21, 24]
```

```
>>> f = my_func(a, b, c)
```

```
>>> f
```

```
[18, 21, 24]
```

```
>>> import math
```

```
>>> math.fsum(f)
63.0
>>>
>>> g = math.fsum(f)
>>> print(g)
63.0
>>>
```

Задание 4.

Исходные данные:

Программа принимает действительное положительное число x и целое отрицательное число y . Выполните возведение числа x в степень y . Задание реализуйте в виде функции **my_func(x, y)**. При решении задания нужно обойтись без встроенной функции возведения числа в степень.

Подсказка: попробуйте решить задачу двумя способами. Первый — возведение в степень с помощью оператора `**`. Второй — более сложная реализация без оператора `**`, предусматривающая использование цикла.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> x = 2
>>> y = -2
>>> a = x ** y
>>> print(a)
0.25
>>>
```

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> x = 2
>>> y = -2
>>> def my_func(x, y):
...     x = float(input("Введите действительное положительное число: "))
```

```

... y = float(input("Введите целое число: "))
... d = pow(x, y)
... return d
...
>>> print(my_func(x, y))
Введите действительное положительное число: 2
Введите целое число: -2
0.25
>>>

```

Задание 5.

Исходные данные:

Программа запрашивает у пользователя строку чисел, разделённых пробелом. При нажатии Enter должна выводиться сумма чисел. Пользователь может продолжить ввод чисел, разделённых пробелом и снова нажать Enter. Сумма вновь введённых чисел будет добавляться к уже подсчитанной сумме.

Но если вместо числа вводится специальный символ, выполнение программы завершается. Если специальный символ введён после нескольких чисел, то вначале нужно добавить сумму этих чисел к полученной ранее сумме и после этого завершить программу.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```

>>> my_list = [25, 55, 77, 88, 99]
>>> import math
>>> math.fsum(my_list)
344.0
>>> a = math.fsum(my_list)
>>> print(a)
344.0
>>> my_list_1 = [88, 65, 85, 25, 48, 92, 74, 29, 99]
>>> import math
>>> math.fsum(my_list_1)
605.0
>>> b = math.fsum(my_list_1)

```

```
>>> print(b)
605.0
>>> c = a + b
>>> print(c)
949.0
>>> my_list_2 = ['abbc']
>>> import math
>>> math.fsum(my_list_2)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: must be real number, not str
>>>
>>> d = math.fsum(my_list_2)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: must be real number, not str
>>>
>>> e = c + d
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'd' is not defined
>>>
>>> print(c)
949.0
>>>
```

Задание 6.

Исходные данные:

Реализовать функцию **int_func()**, принимающую слова из маленьких латинских букв и возвращающую их же, но с прописной первой буквой. Например, **print(int_func('text'))** -> Text.

Решение:

Python 3.8.5 (default, Jan 27 2021, 15:41:15)

[GCC 9.3.0] on linux

Type "help", "copyright", "credits" or "license" for more information.

```
>>> a = "acerbus, administro, ager, albus, ambulo"
```

```
>>> def my_t(a):
```

```
...     b = "acerbus, administro, ager, albus, ambulo"
```

```
...     d = b.title()
```

```
...     return d
```

```
...
```

```
>>> print(my_t(a))
```

```
Acerbus, Administro, Ager, Albus, Ambulo
```

```
>>>
```