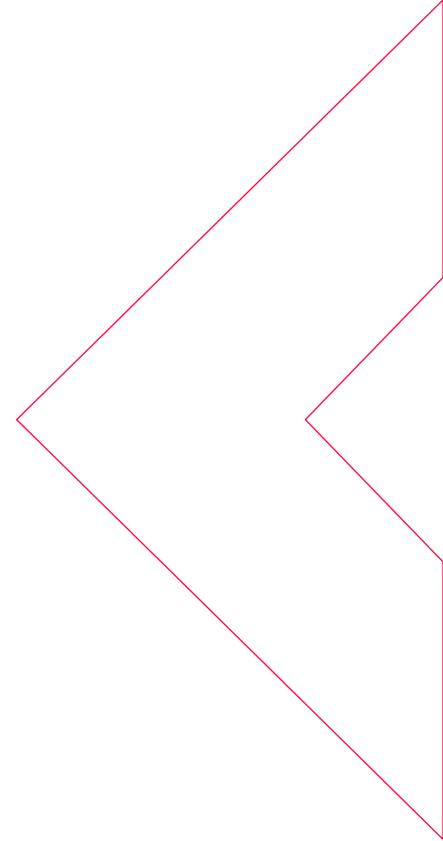


Практические задачи



Задача 1.

Создать программу в среде PyCharm, в которой дан следующий список чисел: **my_list** = [1, 4, 6, 10, 12, 15, 17, 19]. Используя функцию **filter()**, отобрать из данного списка только нечетные числа.

Решение.

Напишем код для решения данной практической задачи и посмотрим на вывод:

main.py	  	Shell 
1	<code>my_list = [1, 4, 6, 10, 12, 15, 17, 19]</code>	Нечетные числа: [1, 15, 17, 19]
2	<code>new_list = list(filter(lambda x: (x%2 == 1), my_list))</code>	>
3	<code>print('Нечетные числа: ', new_list)</code>	
4		

Задача 2.

Написать функцию, которая возвращает заданное пользователем число в четвертую степень, не используя функцию `lambda`.

Решение.

Для решения данной задачи понадобится следующий код:

```
main.py    Shell 
```

```
1 def func(x):  
2     return x * x * x * x  
3  
4 print('Число X в четвертой степени:', func(5))
```

Число X в четвертой степени: 625
> |

Задача 3.

Используя встроенные функции **max()** и **sum()**, найти максимальное значение в последовательности **x = [1, 3, 5, 10, 11]** и посчитать сумму элементов в последовательности **y = [10, 55, 12, 100]**.

Решение.

Напишем код для решения данной практической задачи и посмотрим на вывод:

Python ▾



```
1 x = [1, 3, 5, 10, 11]
2 y = [10, 55, 12, 100]
3 print("Максимальное число в последовательности x: ", max(x))
4 print("Сумма элементов в последовательности y: ", sum(y))
```

Output

Максимальное число в последовательности x: 11

Сумма элементов в последовательности y: 177

Задача 4.

Используя встроенные функции **min()** и **reversed()**, найти минимальное значение в последовательности **a = [-2, 0, 4, 8, 10]** и развернуть порядок элементов в последовательности **b = [6, 4, 2, 0]**.

Решение.

Напишем код для решения данной практической задачи и посмотрим на вывод:

Python ▾



▶ Run

📄 Save

```
1 a = [-2, 0, 4, 8, 10]
2 b = [6, 4, 2, 0]
3 c = reversed(b)
4 print("Минимальное число в последовательности a: ", min(a))
5 print("Обратный порядок следования элементов в последовательности b: ", list(c))
```

Output

```
Минимальное число в последовательности a: -2
Обратный порядок следования элементов в
последовательности b: [0, 2, 4, 6]
```