**

Колледж космического машиностроения и технологии

ОТЧЕТ

По учебной практике УП.01.01 Разработка программных модулей программного обеспечения для компьютерных систем

специальность 09.02.03 Программирование в компьютерных системах

Выполнил студент: Михайлов Данила Алексеевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(подпись)

Оформил студент: Пилипушко Андрей Сергеевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(подпись)

Проверил преподаватель: Гусятинер Леонид Борисович

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(оценка)

Королев, 2020

**Оглавление**

[**Раздел 1. Техника решения задач с использованием структурного и объектно-ориентированного программирования** 3](#_Toc58977522)

[1.1. Выдача заданий на практику 3](#_Toc58977523)

[1.2. Установка интерпретатора Python 3 и настройка окружения 3](#_Toc58977524)

[1.3. Техника работы в командной строке и среде IDLE 5](#_Toc58977525)

[1.4. Техника работы с линейными и разветвляющимися программами 6](#_Toc58977526)

[1.5. Техника работы с циклическими программами, цикл while 10](#_Toc58977527)

[1.6. Техника работы с числами 17](#_Toc58977528)

[1.7. Техника работы со строками 19](#_Toc58977529)

[1.8. Техника работы со списками 21](#_Toc58977530)

[1.9. Техника работы с циклом for и генераторами списков 23](#_Toc58977531)

[1.10. Техника работы с функциями 24](#_Toc58977532)

[1.12. Техника работы с множествами 27](#_Toc58977533)

[1.13. Техника работы с кортежами 28](#_Toc58977534)

[1.14. Техника работы с файлами 29](#_Toc58977535)

[1.15. Техника работы с модулями 31](#_Toc58977536)

[1.16. Техника работы с классами 36](#_Toc58977537)

# **Раздел 1. Техника решения задач с использованием структурного и объектно-ориентированного программирования**

# Выдача заданий на практику

Документы подписаны и лежат на GitHub.

# Установка интерпретатора Python 3 и настройка окружения

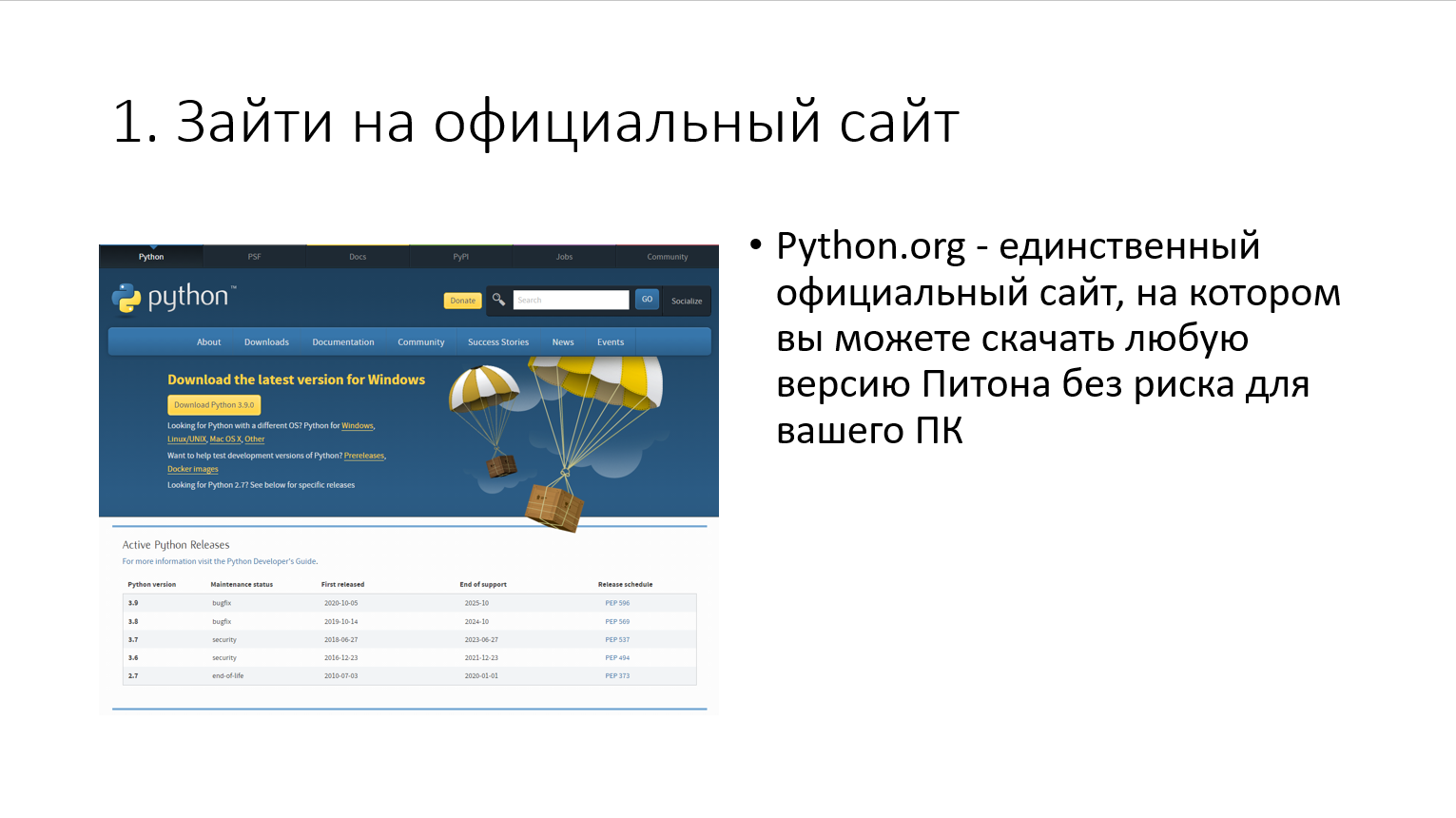


Рис 1. Первый этап

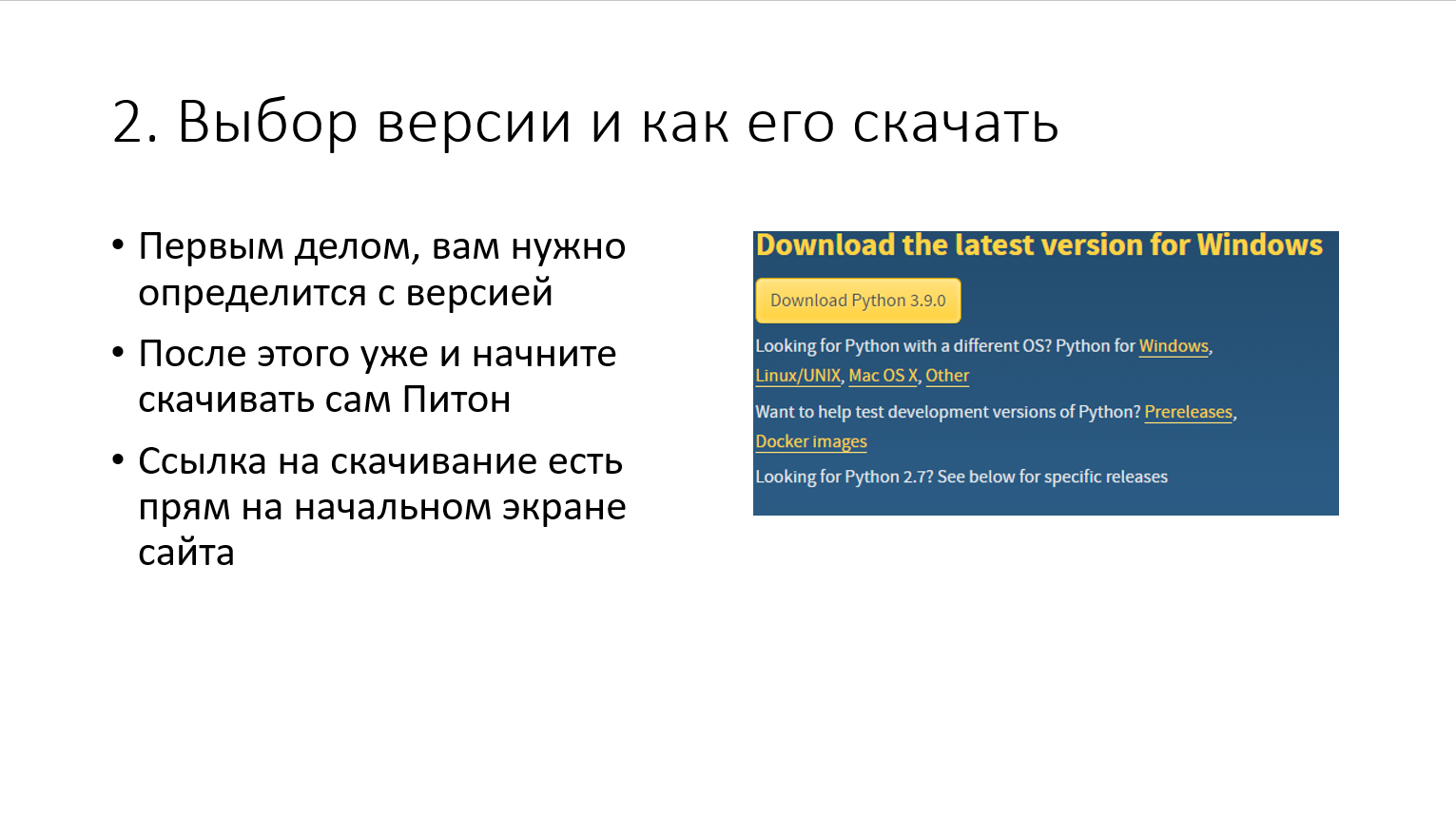


Рис 2. Второй этап

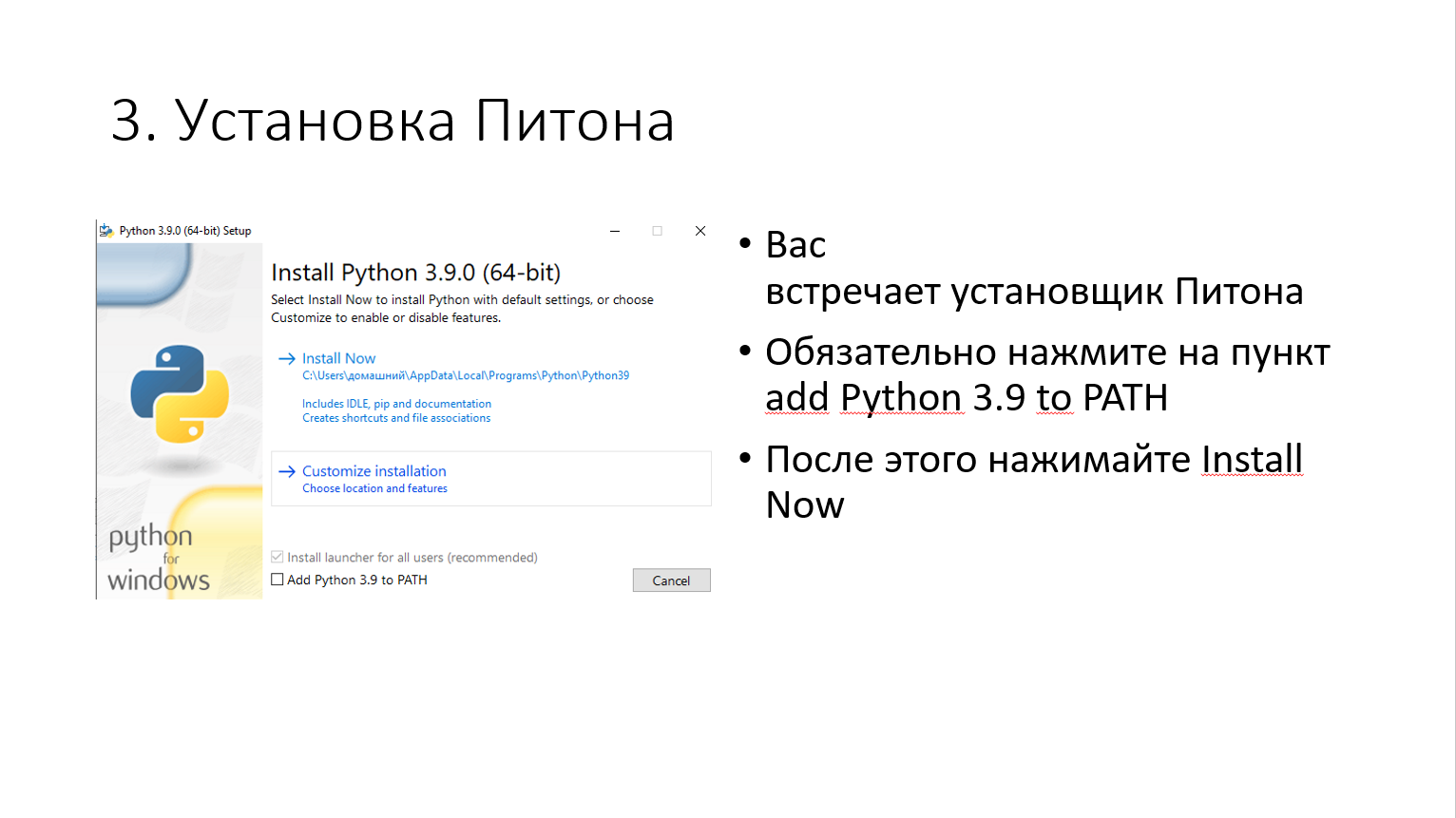


Рис 3. Третий этап.

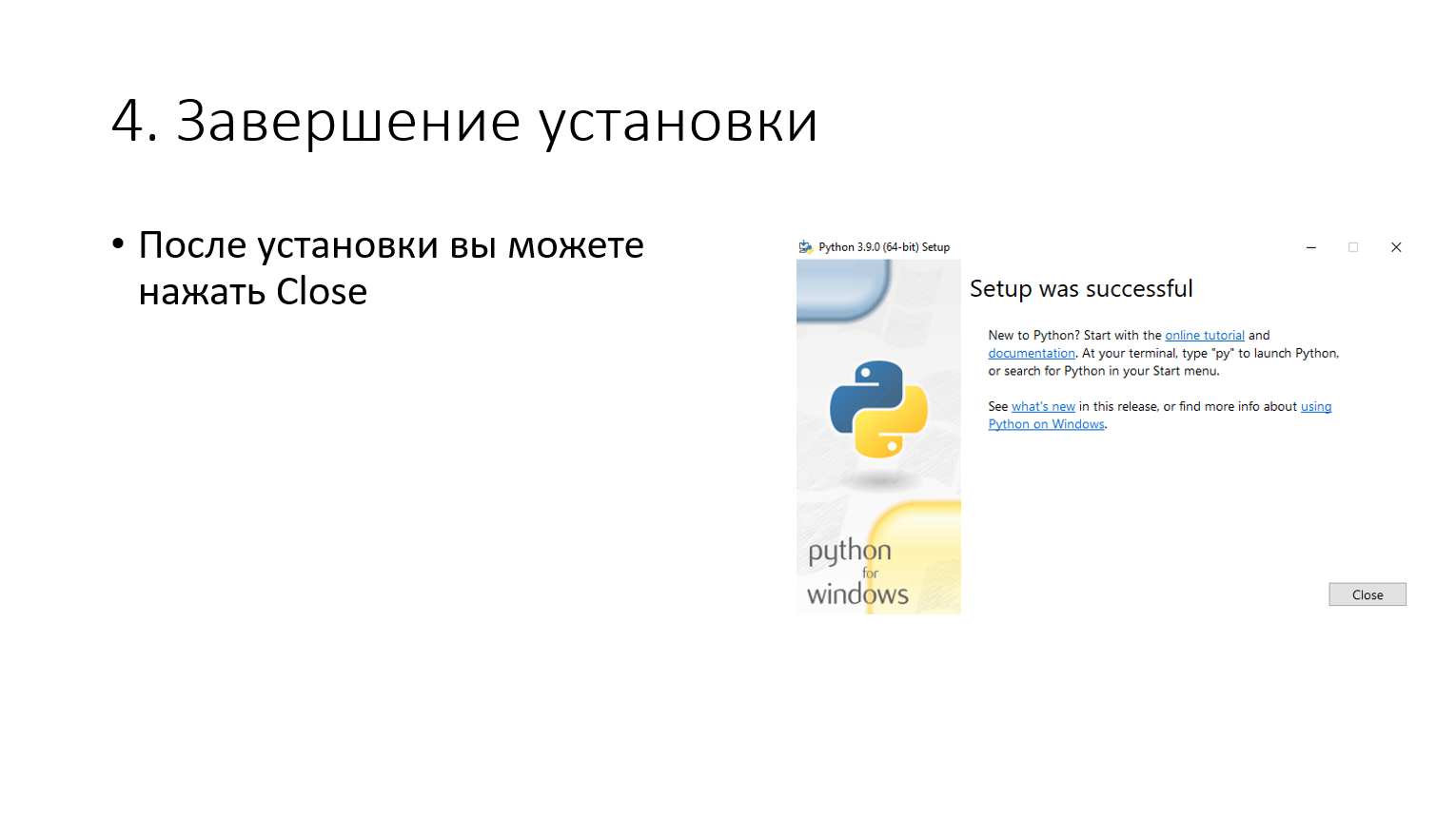


Рис 4. Четвертый этап.

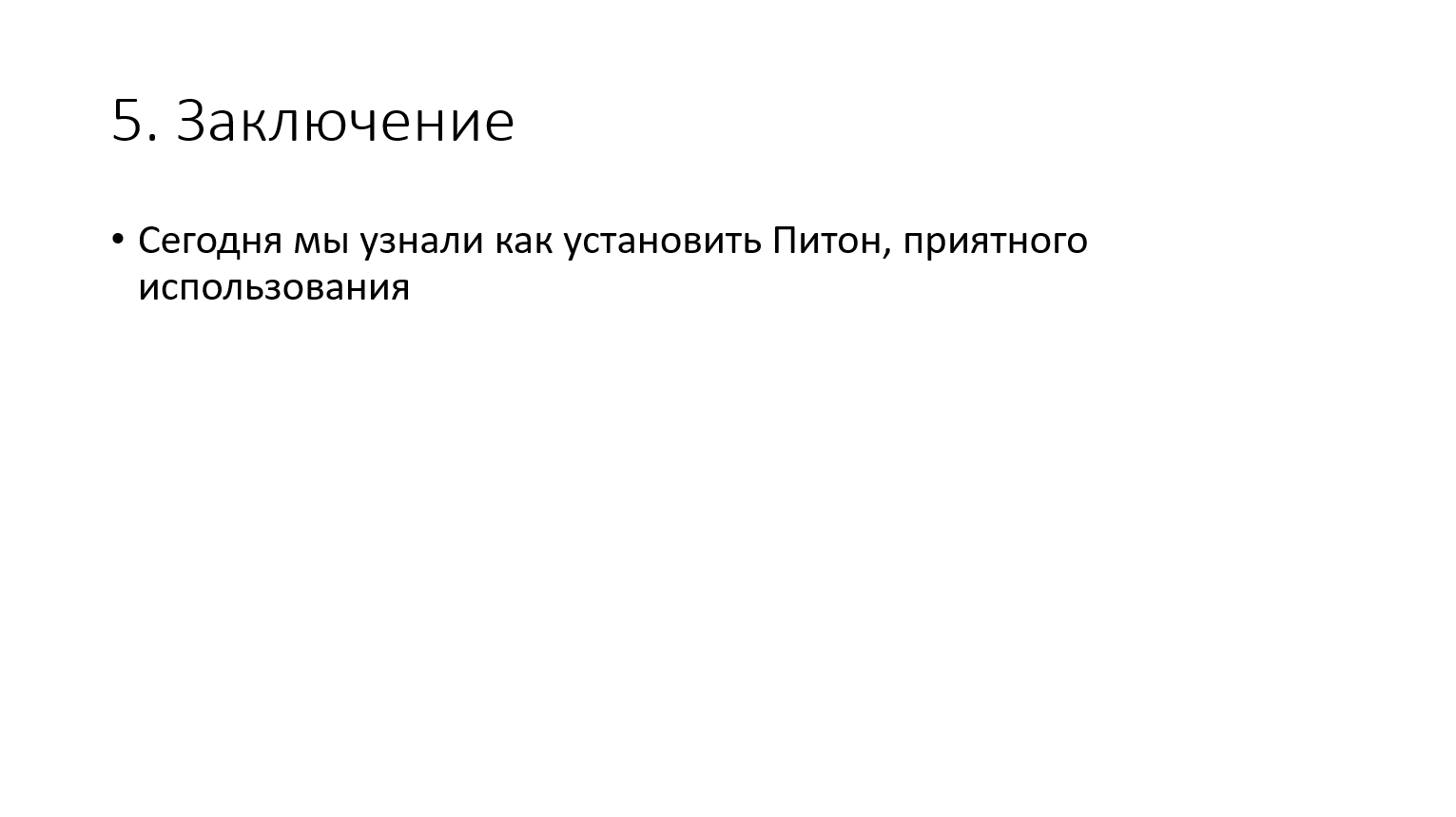


Рис 5. Пятый этап.

# Техника работы в командной строке и среде IDLE

# Техника работы с линейными и разветвляющимися программами

Техника работы с циклическими программами, цикл while

**Задачи на Begin.**

**Задача 1:** (begin1.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Известный книжный магазин "Питонист" обратился к вам как к специалисту за помощью. От вас требуется написать программу, которая принимает на вход две строки: имя автора и название книги (в такой последовательности), а затем выводит небольшое рекламное сообщение от магазина.

name = input()

book = input()

print( name, '- автор бестселлера', book, '- выпустил новую книгу! Спешите приобрести ее в магазине Питонист!')

**Задача 2:** (begin2.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Работник хотел написать программу, которая спрашивает у пользователя дату его рождения (день, месяц и год поочередно) и выводит в логи системы эту дату в формате YYYY - MM - DD. Помогите коллеге и исправьте ошибки.

day = input()

month = input()

year = input()

info = 'Дата рождения пользователя:'

print ( info, year, '-', month, '-', day )

**Задачи на Boolean.**

**Задача 1:** (boolean1.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Дано целое число A.

#Проверить истинность высказывания: «Число A является положительным».

x = int(input("Введите число A: "))

res = x > 0

print(res)

**Задача 2:** (boolean2.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Дано целое число A.

#Проверить истинность высказывания: «Число A является нечетным».

x = int(input("Введите число A: "))

res = x % 2 == 0

print(res)

**Задачи на If**

**Задача 1:** (if1.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Необходимо написать программу, выводящую кортеж, содержащий числа в полуинтервале [a; b), если a < b, и [b; a), если b < a. Числа a и b вводятся с клавиатуры.

a = int(input())

b = int(input())

c = ()

if(a>b):

a,b = b,a

c = tuple(range(a,b))

print(c)

**Задача 2:** (if2.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Даны два целых числа. Выведите значение наименьшего из них.

a = int(input())

b = int(input())

if a < b:

print(a)

else:

print(b)

**Задача 3:** (if3.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Даны два целых числа. Выведите значение наибольшего из них.

a = int(input())

b = int(input())

if a > b:

print(a)

else:

print(b)

**Задачи на Case**

**Задача 1:** (case1.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задание

#Дано целое число в диапазоне 1–7. Вывести строку — название дня недели,

#соответствующее данному числу (1 — «понедельник», 2 — «вторник» и т. д.).

x = int(input("Введите число в диапазоне 1-7: "))

if 0 < x < 8:

if x == 1:

print("Это понедельник!")

elif x == 2:

print("Это вторник!")

elif x == 3:

print("Это среда!")

elif x == 4:

print("Это четверг!")

elif x == 5:

print("Это пятница!")

elif x == 6:

print("Это суббота!")

elif x == 7:

print("Это воскресенье!")

else:

print("Число не в диапазоне 1-7!")

**Задача 2:** (case2.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задание

#Дано целое число K. Вывести строку-описание оценки, соответствующей числу K

#(1 — «плохо», 2 — «неудовлетворительно», 3 — «удовлетворительно», 4 — «хорошо», 5 — «отлично»).

#Если K не лежит в диапазоне 1–5, то вывести строку «ошибка».

x = int(input("Введите K в диапазоне 1-5: "))

if 0 < x < 6:

if x == 1:

print("Плохо")

elif x == 2:

print("Неудовлетворительно")

elif x == 3:

print("Удовретворительно")

elif x == 4:

print("Хорошо")

elif x == 5:

print("Отлично")

else:

print("Ошибка")

**Задача 3:** (case3.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задание

#Элементы окружности пронумерованы следующим образом:

#1 — радиус R, 2 — диаметр D = 2·R, 3 — длина L = 2·π·R, 4 — площадь круга S = π·R2.

#Дан номер одного из этих элементов и его значение.

#Вывести значения остальных элементов данной окружности (в том же порядке).

#В качестве значения π использовать 3.14.

R = float(input("Введите радиус окружности: "))

Pi = float(3.14)

a = int(input("Что хотите вывести?\n1 - Радиус(R)\n2 - Диаметр(D)\n3 - Длина окружности(L)\n4 - Площадь круга(S)\n> "))

if a == 1:

print("Радиус =", R)

elif a == 2:

D = 2 \* R

print("Диаметр =", D)

elif a == 3:

L = 2 \* Pi \* R

print("Длина окружности =", L)

elif a == 4:

S = Pi \* (R \* 2)

print("Площадь круга =", S)

# Техника работы с циклическими программами, цикл while

**Задачи на For.**

**Задача 1:** (for1.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Дано несколько чисел. Вычислите их сумму.

#Сначала вводите количество чисел N, затем вводится ровно N целых чисел.

#Какое наименьшее число переменных нужно для решения этой задачи?

a = [0, int(input())]

for x in range(a[1]):

a[0] += int(input())

print(a[0])

**Задача 2:** (for2.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Дано N чисел: сначала вводится число N, затем вводится ровно N целых чисел.

#Подсчитайте количество нулей среди введенных чисел и выведите это количество.

#Вам нужно подсчитать количество чисел, равных нулю, а не количество цифр.

a = int(input())

b = []

c = 0

for i in range(a):

b.append(int(input()))

for i in range(a):

if b[i-1] == 0:

c += 1

print(c)

**Задача 3:** (for3.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Дано 10 целых чисел. Вычислите их сумму.

#Напишите программу, использующую наименьшее число переменных.

x = 0

for i in range(10):

x += int(input())

print(x)

**Задача 4:** (for4.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#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:

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

print(i, end = ' ')

else:

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

print(i, end = ' ')

**Задача 5:** (for5.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

# 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

n = int(input())

m = 0

for i in range(1, n+1):

m += i

for i in range(n-1):

m -= int(input())

print(m)

**Задачи на While**

**Задача 1:** (while1.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Определите сумму всех элементов последовательности, завершающейся числом 0.

#В этой и во всех следующих задачах числа, следующие за первым нулем,

#учитывать не нужно.

y = 0

while True:

x = int(input())

if x == 0:

break

y += x

print(y)

**Задача 2:** (while2.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#В первый день спортсмен пробежал x километров,

#а затем он каждый день увеличивал пробег на 10% от предыдущего значения.

#По данному числу y определите номер дня,

#на который пробег спортсмена составит не менее y километров.

x, y = int(input()), int(input())

i = 0

while x < y:

x += x / 10

i += 1

print(i+1)

**Задача 3:** (while3.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задача:

#Последовательность состоит из различных натуральных чисел и

#завершается числом 0. Определите значение второго по величине элемента в

#этой последовательности. Гарантируется, что в последовательности есть хотя бы

#два элемента.

a = []

while True:

i = int(input())

a.append(i)

if i == 0:

break

a.sort()

a.pop()

print(a.pop())

**Задача 4:** (while4.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задание:

#Программа получает на вход последовательность целых неотрицательных чисел,

#каждое число записано в отдельной строке.

#Последовательность завершается числом 0,

#при считывании которого программа должна закончить свою работу и

#вывести количество членов последовательности (не считая завершающего числа 0).

#Числа, следующие за числом 0, считывать не нужно.

i = [1, -1]

while i[0] != 0:

i[0] = int(input())

i[1] += 1

print(i[1])

**Задача 5:** (while5.py)

#Выполнил работу Михайлов Данила Алексеевич

#Оформил задачу Пилипушко Андрей Сергеевич

#Задание

#Даны положительные числа A и B (A > B).

#На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений).

#Не используя операции умножения и деления, найти количество отрезков B, размещенных на отрезке A.

a = int(input("Введите А: "))

b = int(input("Введите B: "))

i = 1

while (a > b):

a -= b

i += 1

print("Количество возможных размещенных отрезков B -", i)

**Блок-схема пз 4 и 5**

**Блок Схема к задаче While 5:**

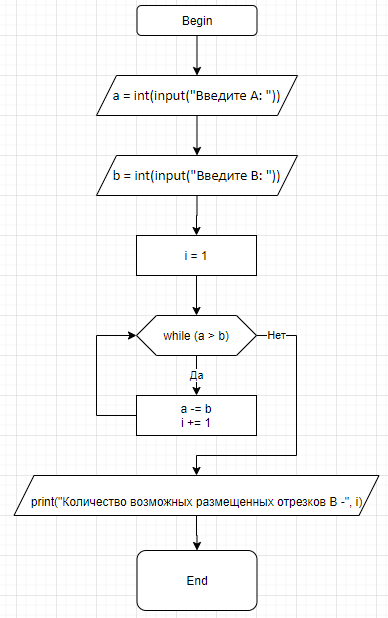


Рис. 1 While 5

**Блок Схема к задаче While 2:**

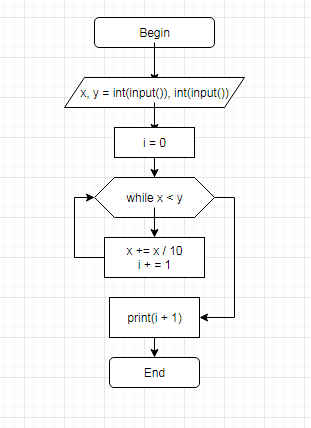


Рис. 2 While 2

**Блок Схема к задаче boolean 1:**



Рис. 3 Boolean 1

**Блок Схема к задаче boolean 2**:

****

Рис. 4 Boolean 2

# Техника работы с числами

**Задача 1**(Задача1.py):

#Выполнил задание Михайлов Данила

#Нахождение длинны отрезка по его концам

a = int(input())

b = int(input())

if a < b:

print(b - a)

elif a > b:

print(a - b)

else:

print('Длинна отрезка равняется 0')

**Задача 2**(Задача2.py)**:**

#Выполнил задание Михайлов Данила

#Нахождение произведения длин отрезков АС и ВС

a, b, c = int(input('Начало отрезка ')), int(input('Конец отрезка ')), int(input('Точка в отрезке '))

print((c - a) \* (b - c))

**Задача 3**(Задача3.py)**:**

#Выполнил задание Михайлов Данила

import math

a = int(input())

s = math.sqrt(3)/4\*a\*\*2

print(s)

**Задача 4**(Задача4.py)**:**

#Выполнил задание Михайлов Данила

#Нахождение корней квадратного уравнения

import math

a, b, c = int(input('A\*x + B\*x + C = 0\nВведите А ')), int(input('В ')), int(input('С '))

d = b\*\*2 - 4\*a\*c

if d > 0:

x = []

x.append((-b - math.sqrt(d)) / 2\*a)

x.append((-b + math.sqrt(d)) / 2\*a)

print('%.1f' % x[0], end = ' ')

print('%.1f' % x[1])

if d == 0:

x = -b / 2\*a

print(x)

if d < 0:

print('Корней нет')

**Задача 5**(Задача5.py)**:**

#Выполнил задание Михайлов Данила

#Нахождение гипотенузы и периметра прямоугольного треугольника

a, b = int(input('Длинна катета а ')), int(input('Длинна катета b '))

print(a + b + (a\*\*2 + b\*\*2)\*(1/2))

# Техника работы со строками

**Задача 1**(Задача1.py)**:**

#Выполнил задание Пилипушко Андрей

#Заменить пробелы в строке на символы нижнего подчеркивания, чтобы строка

#могла сгодиться для логина.

#Если строка состоит из одного слова, менять ничего не нужно.

print(input().replace(' ', '\_'))

**Задача 2**(Задача2.py)**:**

#Задание выполнил Пилипушко Андрей

#Дана строка, состоящая из слов, разделенных пробелами.

#Определите, сколько в ней слов.

print(input().count(' ')+1)

**Задача 3**(Задача3.py)**:**

#Задание выполнил Пилипушко Андрей

#Дана строка.

#Сначала выведите третий символ этой строки.

#Во второй строке выведите предпоследний символ этой строки.

#В третьей строке выведите первые пять символов этой строки.

#В четвертой строке выведите всю строку, кроме последних двух символов.

#В пятой строке выведите все символы с четными индексами (считая, что индексация начинается с 0, поэтому символы выводятся начиная с первого).

#В шестой строке выведите все символы с нечетными индексами, то есть начиная со второго символа строки.

#В седьмой строке выведите все символы в обратном порядке.

#В восьмой строке выведите все символы строки через один в обратном порядке, начиная с последнего.

#В девятой строке выведите длину данной строки.

s = input()

print(s[2])

print(s[-2])

print(s[:5])

print(s[:-2])

print(s[::2])

print(s[1::2])

print(s[::-1])

print(s[::-2])

print(len(s))

**Задача 4**(Задача4.py)**:**

#Выполнил задание Пилипушко Андрей

#https://stepik.org/lesson/201702/step/9?unit=175778

#Уберите точки из введенного IP-адреса.

#Выведите сначала четыре числа через пробел, а затем сумму получившихся чисел.

a = input().split('.')

b = 0

for i in a:

print(int(i), end=' ')

b += int(i)

print('')

print(b)

# Техника работы со списками

**Задача 1**(Задача1.py)**:**

#https://pythontutor.ru/lessons/lists/problems/even\_elements

#Выполнил задание Михайлов Данила

#Выведите все четные элементы списка.

#При этом используйте цикл for, перебирающий элементы списка, а не их индексы!

for num in input().split(' '):

if int(num) % 2 == 0:

print(int(num), end=' ')

**Задача 2**(Задача2.py)**:**

#Выполнил задание Михайлов Данила

#pythontutor.ru

#https://pythontutor.ru/lessons/lists/problems/more\_than\_neighbours/

#Дан список чисел. Определите, сколько в этом списке элементов,

#которые больше двух своих соседей, и выведите количество таких элементов.

#Крайние элементы списка никогда не учитываются, поскольку у них

#недостаточно соседей.

a = [int(n) for n in input().split(' ')]

sch = 0

for i in range(len(a)):

try:

if a[i] > a[i-1] and a[i] > a[i+1] and i != 0 and i != len(a):

sch += 1

except IndexError:

None

print(sch)

**Задача 3**(Задача3.py)**:**

#https://pythontutor.ru/lessons/lists/problems/lineup/

#pythontutor.ru

#Выполнил задание Михайлов Данила

#Петя перешёл в другую школу.

#На уроке физкультуры ему понадобилось определить своё место в строю.

#Помогите ему это сделать.

#Программа получает на вход невозрастающую последовательность натуральных чисел,

#означающих рост каждого человека в строю.

#После этого вводится число X – рост Пети.

#Все числа во входных данных натуральные и не превышают 200.

#Выведите номер, под которым Петя должен встать в строй.

#Если в строю есть люди с одинаковым ростом, таким же, как у Пети,

#то он должен встать после них.

a = [int(n) for n in input().split(' ')]

p = int(input())

for i in range(len(a)):

if a[i] < p:

p = ''

print(i+1)

break

if p != '':

print(len(a) + 1)

# Техника работы с циклом for и генераторами списков

**Задача 1**(Задача1.py)**:**

#Выполнил задание Михайлов Данила

from random import randint

x1 = randint(2, 10)

s = []

for i in range(x1):

r = randint(1,100)

s.append(r)

print(s)

**Задача 2**(Задача2.py)**:**

#Выполнил задание Михайлов Данила

#программа которая выбирает случайные(пронумерованные по сложности задания )

#и создает 7 билетов в которых по 5 заданий

from random import \*

s = ['a1', 'a2', 'a3', 'a4', 'a5', 'a6', 'a7', 'a8', 'a9', 'a10', 'b11', 'b12', 'b13', 'b14', 'b15', 'b16', 'c17',

'c18', 'c19', 'c20']

b = []

gb = []

for i in range(7):

for x in range(5):

e = randint(0,19)

b.append(s[e])

gb.append(billet)

b = []

print(gb)

**Задача 3**(Задача3.py)**:**

#Выполнил задание Михайлов Данила

#Генерирует случайное числа в диапазоне

from random import \*

a = randint(1,20)

d = [i for i in range(a)]

print(d)

**Задача 4**(Задача4.py)**:**

#Выполнил задание Михайлов Данила

#генерирует случайное число. Если i делится на 3 без остатка, то пропускает его

from random import \*

a = randint(0, 150)

d = [i\*\*2 for i in range(a) if i % 3 != 0]

print(d)

# Техника работы с функциями

**Задача 1**(func1.py)**:**

# http://ptaskbook.com/ru/tasks/func.php

# Выполнил задание Михайлов Данила

'''

Описать функцию SumRange(A, B) целого типа, находящую сумму всех целых

чисел от A до B включительно (A и B — целые).

Если A > B, то функция возвращает 0.

С помощью этой функции найти суммы чисел от A до B и от B до C,

если даны числа A, B, C.

'''

def SumRange(x, y):

s = 0

if x < y:

for i in range(x, y+1):

s += i

return s

elif x > y:

return 0

a, b, c = int(input()), int(input()), int(input())

print(SumRange(a, b))

print(SumRange(b, c))

**Задача 2**(func2.py)**:**

# http://ptaskbook.com/ru/tasks/func.php

# Выполнил задание Михайлов Данила

'''

Описать функцию IsSquare(K) логического типа, возвращающую True,

если целый параметр K (> 0) является квадратом некоторого целого числа, и False

в противном случае. С ее помощью найти количество квадратов в наборе из 10 целых

положительных чисел.

'''

import math

def IsSquare(x):

if math.sqrt(x).is\_integer():

return True

else:

return False

a = [int(n) for n in input('Введите 10 чисел через пробел: ').split() if IsSquare(int(n))]

print(\*a)

* 1. Техника работы со словарями

**Задача 1**(dict1.py)**:**

#Есть имена учеников и номер их зачетной книжки, нужно занести все в словарь

n = ['Пилипушко', 'Михайлов']

num = [1,2]

a = {}

for i in range(len(n)):

a.update({num[i]:n[i]})

print(a)

**Задача 2**(dict2.py)**:**

# Выполнил задание Михайлов Данила

'''

Условие. В единственной строке записан текст. Для каждого слова из данного текста

подсчитайте, сколько раз оно встречалось в этом тексте ранее.

Словом считается последовательность непробельных символов идущих подряд, слова разделены

одним пробелом или символами конца строки.

'''

a = {}

sch = 1

for i in input().split():

if i not in a:

a.update({i:sch})

else:

a.update({i:a[i] + 1})

print(a)

**Задача 3**(dict3.py)**:**

# https://pythontutor.ru/lessons/sets/problems/number\_of\_words/

# Выполнил задание Михайлов Данила

'''

Условие. Дан текст: в первой строке задано число строк, далее идут сами строки.

Выведите слово, которое в этом тексте встречается чаще всего. Если таких слов несколько,

выведите все.

'''

a = {}

for i in range(int(input())):

st = input()

if st not in a:

a.update({st:1})

else:

a.update({st:a[st] + 1})

sch = max(a.values())

for k, v in a.items():

if v == sch:

print(k + ':', v)

# Техника работы с множествами

**Задача 1**(задание1.py)**:**

# https://pythontutor.ru/lessons/sets/problems/number\_of\_unique/

# Выполнил задание Михайлов Данила

'''

Задача «Количество различных чисел»

Условие. Дан список чисел. Определите,

сколько в нем встречается различных чисел.

'''

print(len(set(input().split())))

**Задача 2**(задание2.py)**:**

# https://pythontutor.ru/lessons/sets/problems/number\_of\_coincidental/

# Выполнил задание Михайлов Данила

'''

Даны два списка чисел. Посчитайте, сколько чисел содержится одновременно как

в первом списке, так и во втором.

'''

print(len(set(input().split()) & set(input().split())))

# Техника работы с кортежами

**Задача 1**(задание1.py)**:**

# https://stepik.org/lesson/193753/step/4?unit=168148

# Выполнил задание Михайлов Данила

'''

Вывести чётные

Необходимо вывести все четные числа на отрезке [a; a \* 10].

'''

a = int(input())

[print(i) for i in range(a, a\*10+1) if i % 2 == 0]

**Задача 2**(задание2.py)**:**

# https://stepik.org/lesson/193753/step/5?unit=168148

# Выполнил задание Михайлов Данила

'''

С клавиатуры вводятся целые числа a > b. Выведите убывающую последовательность чисел

по одному числу в строке.

'''

a, b = int(input()), int(input())

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

print(i)

# Техника работы с файлами

**Задача 1**(text1.py)**:**

# http://ptaskbook.com/ru/tasks/text.php

# Выполнил задание Михайлов Данила

'''

Дана строка S и текстовый файл. Добавить строку S в конец файла

'''

f = open("text1.txt", "r")

a = f.read()

a += 'S'

f.close()

f = open("text1.txt", "w")

f.write(a)

f.close()

**Задача 2**(text2.py)**:**

# http://ptaskbook.com/ru/tasks/text.php

# Выполнил задание Михайлов Данила

'''

Дана строка S и текстовый файл. Заменить в файле все пустые строки на строку S.

'''

f = open("text2.txt", "r")

a = ''

for i in f.read():

for j in i:

if j == ' ':

j = 'S'

a += j

f.close()

f = open("text2.txt", "w")

f.write(a)

f.close()

**Задача 3**(text3.py)**:**

# http://ptaskbook.com/ru/tasks/text.php

# Выполнил задание Михайлов Данила

'''

Дан текстовый файл. Заменить в нем все подряд идущие пробелы на один пробел.

'''

f = open("text3.txt", "r")

a = ''

sch = 0

for i in f.read():

for j in i:

if sch == 1 and j == ' ':

j = ''

elif j != ' ':

sch = 0

if j == ' ':

sch = 1

a += j

f.close()

f = open("text3.txt", "w")

f.write(a)

f.close()

**Задача 4**(text4.py)**:**

# http://ptaskbook.com/ru/tasks/text.php

# Выполнил задание Михайлов Данила

'''

Дан текстовый файл, каждая строка которого изображает целое число, дополненное слева и справа

несколькими пробелами. Вывести количество этих чисел и их сумму.

'''

f = open("text4.txt", "r")

a = [n for n in f.read()]

b = [int(i) for i in a if i != ' ' and i != '\n']

print(len(b), sum(b))

f.close()

**Задача 5**(text5.py)**:**

# http://ptaskbook.com/ru/tasks/text.php

# Выполнил задание Михайлов Данила

'''

Дан текстовый файл. Создать символьный файл, содержащий все знаки препинания,

встретившиеся в текстовом файле (в том же порядке).

'''

f = open("text5.txt", "r")

a = [[j for j in i if j.islower() == 0 and j.isdigit() == 0 and j.isalpha() == 0 and j.isupper() == 0 and j != ' ' and j != '\n'] for i in f.read()]

[print(\*i) for i in a if i]

f.close()

# Техника работы с модулями

**Задание 1(15-1.1):**

# Выполнил задание Михайлов Данила

# Контейнерные типы данных модуля collections.

# https://docs-python.ru/standart-library/modul-collections-python/

# Класс deque() модуля collections в Python.

# https://docs-python.ru/standart-library/modul-collections-python/klass-deque-modulja-collections/

# Двусторонняя очередь в Python.

# Класс collections.deque() это обобщение стеков и очередей и представляет собой двустороннюю очередь.

# Двусторонняя очередь deque() поддерживает поточно-ориентированные, эффективные по памяти операции добавления

# и извлечения элементов последовательности с любой стороны с примерно одинаковой производительностью O(1)

# в любом направлении.

#

# Списки поддерживают аналогичные операции, но они оптимизирован только для быстрых операций с последовательностями

# фиксированной длины и требуют затрат O(n) на перемещение памяти для операций pop(0) и insert(0, v), которые изменяют

# как размер, так и положение базового представления данных.

from collections import deque

d = deque(["One", "Two", "Three"])

print(d)

d.append("Four")# Добавим вправо

print(d)

d.appendleft("Zero")# Добавим влево

print(d)

d.pop()# Убираем справа

print(d)

d.popleft()# Убираем слева

print(d)

d.reverse()# Разворот

print(d)

d = deque([], maxlen=3)# Ограничение размера

d.append(1)

print(d)

d.append(12)

print(d)

d.append(123)

print(d)

d.append(1234)

print(d)

**Задание 1(15-1.2):**

# Выполнил задание Михайлов Данила

# Контейнерные типы данных модуля collections.

# Класс Counter() модуля collections в Python.

# https://docs-python.ru/standart-library/modul-collections-python/klass-counter-modulja-collections/

# Подсчет количества повторений элементов в последовательности.

# класс collections.Counter() предназначен для удобных и быстрых подсчетов количества появлений неизменяемых

# элементов в последовательностях.

#

# from collections import Counter

# c = Counter(['a', 'b', 'a', 'c', 'b', 'a'])

# print(dict(c))

# Вывод => {'a': 3, 'b': 2, 'c': 1}

import collections

c = (['a', 'b', 'a', 'c', 'c', 'b', 'a', 'c', 'a', 'a', 'b'])

print(c)# Вывод списка

print(collections.Counter(c))# Вывод кол-ва каждых элементов в списке

**Задание 2(15-2):**

# Выполнил задание Михайлов Данила

# Варианты использования defaultdict

from collections import defaultdict

a = [('a', 1),

('b', 2),

('c', 3),

('a', 4),

('c', 5)]

b = {}

for k, v in a:

b.setdefault(k, []).append(v)

sorted(b.items())

print(a)

print(b)

print()###

s = [('a', 1),

('b', 2),

('b', 3),

('a', 4),

('a', 5),

('b', 6)]

d = defaultdict(set)

for k, v in s:

d[k].add(v)

sorted(d.items())

print(d)

print()###

def c(a):

return lambda: a

d = defaultdict(c('<missing>'))

d.update(name='Micke', action='kick')

print(d)

**Задание 4(15-4):**

# Выполнил задание Михайлов Данила

#

# Работа с файлами в Python с помощью модуля OS

#

# Задание 1. Вывод текущей директории

# Задание 2. Создание папки

# Задание 3. Изменение директории

# Задание 4. Создание вложенных папок

# Задание 5. Создание файлов

# Задание 6. Переименование файлов

# Задание 7. Перемещение файлов

# Задание 8. Список файлов и директорий

# Задание 9. Удаление файлов

# Задание 10. Удаление директорий

# Задание 11. Получение информации о файлах

import os # Библиотека для работы с файлами

import shutil

# Задание 1. Вывод текущей директории

print('Задание 1.')

print('Текущая деректория:', os.getcwd())

# Задание 2. Создание папки

print('Задание 2.')

fname = input('Введите имя файла: ')

'''

if not os.path.isdir(name):

os.mkdir(name)

'''

# Задание 3. Изменение директории

print('Задание 3.')

chname = input('Введите имя файла для изминения деректории: ')

os.chdir(chname)

print('Текущая директория изменилась на <', chname, '> :', os.getcwd())

# Задание 4. Создание вложенных папок

print('Задание 4.')

nfile = input('Введите имя конечного файла: ')

os.makedirs(nfile)

# Задание 5. Создание файлов

print('Задание 5.')

createFileName = input('Введите имя создоваемого файла с форматом: ')

textFile = open(createFileName, 'w') # Создать новый текстовый файл

createText = input('Введите содержание файла: ')

textFile.write(createText) # запить текста в этот файл

# Задание 6. Переименование файлов

print('Задание 6.')

os.rename('name.txt', 'rename.txt')

# Задание 7. Перемещение файлов

print('Задание 7.')

shutil.move('Путь до файла', 'Новый путь до файла')

# Задание 8. Список файлов и директорий

print('Задание 8.')

os.listdir('Путь до папки')

# Задание 9. Удаление файлов

print('Задание 9.')

os.remove('Путь до файла')

# Задание 10. Удаление директорий

print('Задание 10.')

os.rmdir('Путь до файла')

# Задание 11. Получение информации о файлах

print('Задание 11.')

print(os.stat('name.txt'))

# Техника работы с классами