



Государственное бюджетное образовательное учреждение высшего образования  
Московской области

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ**

---

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

Выполнили студенты:

Джабраилов Тимур Адамович

Завадский Михаил Андреевич

Толоконников Алексей Михайлович

\_\_\_\_\_ (подпись)

\_\_\_\_\_ (подпись)

\_\_\_\_\_ (подпись)

Проверил преподаватель:

Гусятинер Леонид Борисович

\_\_\_\_\_ (подпись)

\_\_\_\_\_ (оценка)

Королев, 2021

## Содержание

Практическое занятие №2.....	2
Практическое занятие №3.....	3
Практическое занятие №4.....	5
Практическое занятие №5.....	10
Практическое занятие №6.....	13
Практическое занятие №7.....	15
Практическое занятие №8.....	17
Практическое занятие №9.....	18
Практическое занятие №10.....	19
Практическое занятие №11.....	20
Практическое занятие №12.....	21
Практическое занятие №13.....	22
Практическое занятие №14.....	23
Практическое занятие №15.....	24
Практическое занятие №16.....	25

## Практическое занятие №2

Обучающий ролик по установке питона: <https://yadi.sk/i/L4ApJ8CK3ns7Ow>

Материал из ролика:

ссылка на скачивание: <https://www.python.org/downloads/windows/>

Текстовый вариант:

1. Заходим на сайт <https://www.python.org/downloads/>.
2. Нажимаем на кнопку “Download Python 3.9.0”.
3. Ожидаем скачивания программы.
4. Запускаем, ставим галочку “Add Python 3.9 to PATH” и начинаем установку.
5. Дожидаемся установки и начинаем работать с Python.

## Практическое занятие №3

Для начала работы откроем командную строку

Что-бы открыть командную строку нужно нажать комбинацию клавиш: Win+ R

После чего нужно ввести слово (cmd) для открытия командной строки

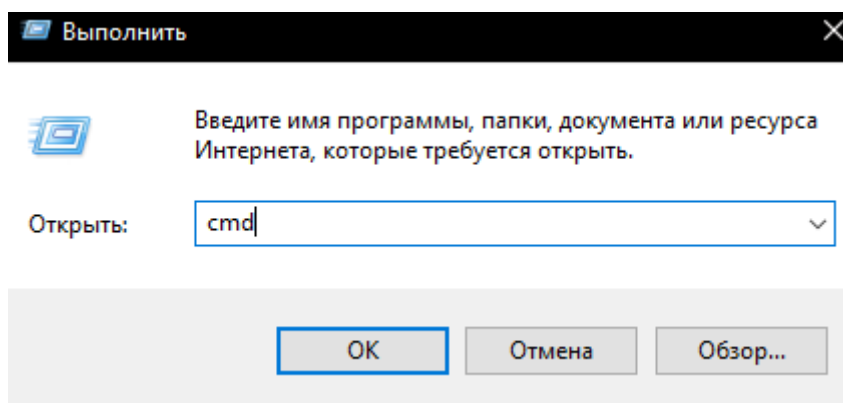


рис.1

После выполнения команды появляется пустая командная строка рис.2

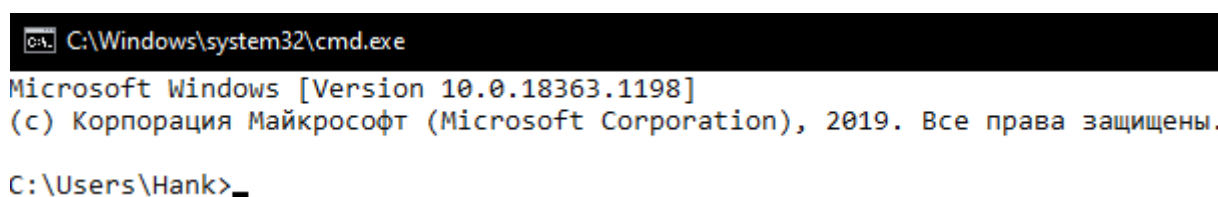


рис.2

Изначально командная строка открывает папку с пользователем, что бы перейти в корневую папку диска нужно ввести данную команду:

```
C:\Users\Hank>cd C:\  
C:\>_
```

рис.3

Но часто командная строка не даёт как-либо взаимодействовать с корневой папкой диска, поэтому создаем новую папку

```
C:\>mkdir new_folder
```

рис.4

Мы создали новую папку

После чего нам нужно перейти в неё

```
C:\>cd C:\new_folder\
```

Рис.5

Теперь мы можем создать файл с кодом

Для этого введем следующую команду

```
C:\new_folder>copy con new_file.py
```

Рис.6

Мы создали новый пустой файл с расширением python

Далее мы можем ввести код

После того как мы закончим с написанием кода, нажмем на клавишу Enter, после чего нажмем на клавишу f6 и Enter

```
C:\new_folder>copy con new_file.py
print("Hello World !")
^Z
Скопировано файлов:          1.
```

Рис.7

По итогу мы видим, что был создан (скопирован) данный файл

Давайте запустим его

```
C:\new_folder>new_file.py
Hello World !

C:\new_folder>_
```

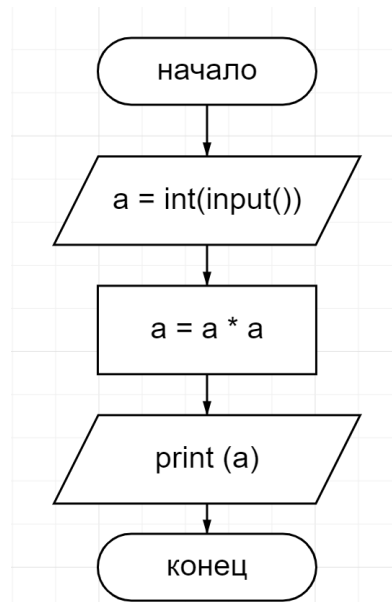
Рис.8

В итоге мы видим, что программа запустилась и вывела код

## Практическое занятие №4

### Задача №1(Begin)

```
# Выполнил Завадский Михаил П2-18.  
# Дана сторона квадрата а. Найти его периметр  $P = 4 \cdot a$ .  
a = int(input())  
a = a * 4  
print(a)
```



блок –схема 1

### Задача №2(Begin)

```
# Выполнил Завадский Михаил П2-18.  
# Дана сторона квадрата а. Найти его площадь  $S = a^2$ .  
a = int(input())  
a = a * a  
print(a)
```

### Задача №1(Bool)

```
# Выполнил Завадский Михаил П2-18.  
# Дано целое число А.  
# Проверить истинность высказывания: «Число А является  
# положительным».  
x = int(input())  
x = x > 0  
print(x)
```

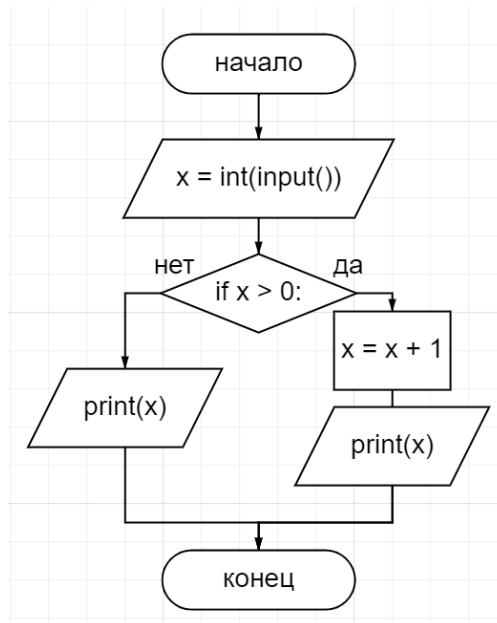
### Задача №2(Bool)

```
# Выполнил Завадский Михаил П2-18.  
# Дано целое число А.  
# Проверить истинность высказывания: «Число А является
```

```
# нечетным».
a = int(input())
a = a % 1 == 0
print(a)
```

### Задача №1(if)

```
# Сделал Толоконников А. М. П2-18.
# Дано целое число. Если оно является положительным, то
# прибавить к нему 1;
# в противном случае не изменять его. Вывести полученное число.
x = int(input())
if x > 0:
    x = x + 1
    print(x)
else:
    print(x)
```



**блок –схема 2**

### Задача №2(if)

```
# Сделал Толоконников А. М. П2-18.
# Дано целое число. Если оно является положительным, то
# прибавить к нему 1;
# в противном случае вычесть из него 2. Вывести полученное число.
x = int(input())
if x > 0:
    x = x + 1
    print(x)
else:
    x = x - 2
    print(x)
```

### Задача №3(if)

```
# Сделал Джабраилов Тимур П2-18.
# Дано целое число. Если оно является положительным, то
# прибавить к нему 1;если
# отрицательным, то вычесть из него 2;если нулевым, то заменить
# его на 10.
# Вывести полученное число
x = int(input())
if x > 0:
    x = x + 1
    print(x)
elif x < 0:
    x = x - 2
    print(x)
else:
    x = 10
    print(x)
```

### Задача №4(if)

```
# Сделал Толоконников А. М. П2-18
# Дано 10 целых чисел. Вычислите их сумму. Напишите программу,
# использующую наименьшее число переменных.
```

```
sum = 0

for i in range(10):
    n = int(input())
    sum+= n
print(sum)
```

### Задача №5(if)

```
# Сделал Толоконников А. М. П2-18
# По данному натуральному n вычислите сумму  $1^3 + 2^3 + 3^3 + \dots + n^3$ .
#  $1^3 + 2^3 + \dots + n^3$ .
n = int(input())
sum = 0
for i in range(1, n + 1):
    sum += i ** 3
print(sum)
```

### Задача №1(case)

```
# Выполнил Завадский Михаил П2-18.
# 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())
```



```

if 0 < x < 8:
    if x == 1:
        print("Понедельник")
    elif x == 2:
        print("Вторник")
    elif x == 3:
        print("Среда")
    elif x == 4:
        print("Четверг")
    elif x == 5:
        print("Пятница")
    elif x == 6:
        print("Суббота")
    elif x == 7:
        print("Воскресенье")
else:
    print("Неправильное число")

```

### Задача №2(case)

```

# Выполнил Джабраилов Тимур П2-18.
# Дано целое число К. Вывести строку-описание оценки,
# соответствующей числу К(1 – «плохо», 2 –
# «неудовлетворительно», 3 – «удовлетворительно», 4 – «хорошо»,
# 5 – «отлично»). Если К не лежит в диапазоне 1-5, то вывести
# строку «ошибка».
x = int(input())
if 0 < x < 6:
    if x == 1:
        print("Плохо")
    elif x == 2:
        print("Неудовлетворительно")
    elif x == 3:
        print("Удовлетворительно")
    elif x == 4:
        print("Хорошо")
    elif x == 5:
        print("Отлично")
else:
    print("Ошибка")

```

### Задача №3(case)

```

# Сделал Толоконников А. М.П2-18.
# Дано целое число. Если оно является положительным, то
# прибавить к нему 1;если
# отрицательным, то вычесть из него 2;если нулевым, то заменить
# его на 10.
# Вывести полученное число
x = int(input())

```

```
if x > 0:
    x = x + 1
    print(x)
elif x < 0:
    x = x - 2
    print(x)
else:
    x = 10
    print(x)
```

## Практическое занятие №5

### Задача №1(for)

# Сделал Толоконников А. М. П2-18  
# Даны два целых числа А и В (при этом  $A \leq B$ ). Выведите все  
# числа от А до В включительно.

```
a = int(input())
b = int(input())
for i in range(a, b+1):
    print(i)
```

### Задача №2(for)

# Сделал Толоконников А. М. П2-18  
# Даны два целых числа А и В. Выведите все числа от А до В  
# 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):
        print(i)
else:
    for i in range(a, b - 1, - 1):
        print(i)
```

### Задача №3(for)

# Сделал Толоконников А. М. П2-18  
# Даны два целых числа А и В,  $A > B$ . Выведите все нечётные числа  
# от  
# А до В 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=' ')
```

### Задача №4(for)

#Сделал Толоконников А. М. П2-18  
#Дано 10 целых чисел. Вычислите их сумму. Напишите программу,  
использующую наименьшее число переменных.

```
sum = 0
```

```

for i in range(10):
    n = int(input())
    sum+= n
print(sum)

```

### Задача №5(for)

# Сделал Толоконников А. М. П2-18  
 # По данному натуральному n вычислите сумму  $1^3 + 2^3 + 3^3 + \dots + n^3$ .

```

n = int(input())
sum = 0
for i in range(1, n + 1):
    sum += i ** 3
print(sum)

```

### Задача №1(while)

# Сделал Толоконников А. М. П2-18.  
 # По данному целому числу N распечатайте все квадраты натуральных чисел, не превосходящие N, в порядке возрастания.

```

n = int(input())
i = 1
while i ** 2 <= n:
    print(i ** 2)
    i += 1

```

### Задача №2(while)

# Сделал Толоконников А. М. П2-18.  
 # Дано целое число, не меньшее 2.  
 # Выведите его наименьший натуральный делитель, отличный от 1.

```

n = int(input())
i = 2
while n % i != 0:
    i += 1
print(i)

```

### Задача №3(while)

# Сделал Толоконников А. М. П2-18.  
 # По данному натуральному числу N найдите наибольшую целую степень двойки, не превосходящую N.  
 # Выведите показатель степени и саму степень.

```

n=int(input())
a = 2
i = 1
while a <= n:

```

```
    a *= 2
    i += 1
print(i - 1, a // 2)
```

### **Задача №4(while)**

```
# Выполнил Завадский Михаил П2-18.
# 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.
```

```
len = 0
while int(input()) != 0:
    len += 1
print(len)
```

### **Задача №5(while)**

```
# Выполнил Завадский Михаил П2-18.
# Определите сумму всех элементов последовательности,
# завершающейся числом 0.
# В этой и во всех следующих задачах числа, следующие за первым
# нулем, учитывать не нужно.
```

```
sum = 0
element = int(input())
while element != 0:
    sum += element
    element = int(input())
print(sum)
```

## Практическое занятие №6

### Задача по модулю fraction

```
# Завадский Михаил Задание разбор модуля Fraction
# Модуль fractions

from fractions import Fraction
print (Fraction())
# по умолчанию
(Fraction(0, 1))
print (Fraction(2, 4), Fraction(3, 9), Fraction(4, 16))
# если есть общий делитель то, числа будут сокращены
print (Fraction('1/3').__round__())
# округляет до ближайшего чётного числа
print (Fraction(1, 3) + Fraction(1, 4))
# также можно выполнять разные математические действия
print (Fraction(1, 3) - Fraction(1, 4))
print (Fraction(1, 3) ** Fraction(1, 4))
print (Fraction(1, 3) % Fraction(1, 4))
print (Fraction(1, 3) * Fraction(1, 4))
```

### Задача по модулю decimal

```
# Сделал Толоконников Алексей Михайлович П2-18
# Decimal- вычисления с заданной точностью
from decimal import Decimal
# Округление чисел
# Объекты Decimal имеют метод quantize(), который
# позволяет округлять числа.
# В этот метод в качестве первого аргумента передается
# также объект Decimal,
# который указывает формат округления числа:

number = Decimal("0.444")
number = number.quantize(Decimal("1.00"))
print(number)

number = Decimal("0.555678")
print(number.quantize(Decimal("1.00")))

number = Decimal("0.999")
print(number.quantize(Decimal("1.00")))

# Как отделить рубли от копеек в вещественных числах?
# Decimal вместо float:

x = Decimal('100.25')
```

```
a = int(x)
b = int(100 * (x - a))
print(a, b)
```

### **Задача по модулю complex**

```
# Сделали Толоконников Алексей Михайлович, Завадский Михаил
# Андреевич
# Модуль Complex

a = complex(10, 2)
print(a)
b = complex(6, 4)
print(b)
c = a + b
print(c)
print(a.imag) # Мнимая часть
print(b.real) # Действительная часть
print(a == b) # Проверка равенства
print(abs(a)) # Модуль комплексного числа
print(pow(b, 5)) # Возведение в пятую степень комплексного числа
```

## Практическое занятие №7

### Задача №1(stroka)

```
# Сделал Толоконников Алексей Михайлович
# Дана строка, состоящая из слов, разделенных пробелами.
# Определите, сколько в ней слов.
# Используйте для решения задачи метод count.
```

```
s = input()
print(s.count(' ') + 1)
```

### Задача №2(stroka)

```
# Сделал Толоконников Алексей Михайлович
# Дана строка. Разрежьте ее на две равные части (если длина
# строки — четная,
# а если длина строки нечетная, то длина первой части должна
# быть на один символ больше).
# Переставьте эти две части местами, результат запишите в новую
# строку и выведите на экран.
# При решении этой задачи не стоит пользоваться инструкцией if.
```

```
s = input()
l = len(s)//2+len(s)%2
a = s[l:]
b = s[:l]
print(a+b)
```

### Задача №3(stroka)

```
# Сделал Толоконников Алексей Михайлович
# Дана строка, состоящая ровно из двух слов, разделенных
# пробелом.
# Переставьте эти слова местами. Результат запишите в строку и
# выведите получившуюся строку.
# При решении этой задачи не стоит пользоваться циклами и
# инструкцией if.
```

```
s = input()
first_word = s[:s.find(' ')]
second_word = s[s.find(' ') + 1:]
print(second_word + ' ' + first_word)
```

### Задача №4(stroka)

```
# Сделал Толоконников Алексей Михайлович
# 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.
```

```
a = input()  
print(a[2])  
print(a[-2])  
print(a[0:5])  
print(a[0:-2])  
print(a[::2])  
print(a[1::2])  
print(a[::-1])  
print(a[::-2])  
print(len(a))
```

## Практическое занятие №8

### Задача №1(список)

```
# исполнитель Завадский Михаил
# Дан список чисел. Выведите значение наибольшего элемента в
# списке,
# а затем индекс этого элемента в списке.
# Если наибольших элементов несколько, выведите индекс первого
# из них.
a = input().split()
n = 0
for i in range(len(a)):
    a[i]=int(a[i])
for i in range(1,len(a)-1):
    if a[i]>a[i-1] and a[i]>a[i+1]:
        n+=1
print(n)
```

### Задача №2(список)

```
# Выполнил Толоконников Алексей Михайлович
# Дан список чисел. Посчитайте, сколько в нем пар элементов,
# равных друг другу. Считается, что любые два элемента,
# равные друг другу образуют одну пару, которую необходимо
# посчитать.
```

```
a = [int(l) for l in input().split()]
m = min(a)
M = max(a)
n1, n2, n3, y = 0,0,0,0
for i in range(m, M+1):
    x = a.count(i)
    for l in range(0, x):
        n2 = n3
        n1 = l
        n3 = n1 + n2
    y += n3
    n1, n2, n3 = 0,0,0
print(y)
```

## Практическое занятие №9

```
# Выполнил Толоконников Алексей Михайлович П2-18.  
# Дана квадратная матрица порядка М. Обнулить элементы матрицы,  
# лежащие ниже главной диагонали. Условный оператор не  
# использовать.
```

```
import random as rnd
```

```
m = rnd.randrange(2, 10, 2)  
n = m  
z = [rnd.randrange(0, 100) for i in range(m)]  
print('Матрица размером:', m, 'на', m)  
x = [z] * n  
print(*x, sep='\n')
```

## Практическое занятие №10

```
# Выполнил Толоконников Алексей Михайлович П2-18.  
# Даны четыре действительных числа: x1, y1, x2, y2.  
# Напишите функцию distance(x1, y1, x2, y2), вычисляющая  
# расстояние между точкой (x1,y1) и (x2,y2).  
# Считайте четыре действительных числа и выведите результат работы  
# этой функции.
```

```
from math import sqrt
```

```
def distance(x1, y1, x2, y2):
```

```
    return sqrt((x1 - x2) ** 2 + (y1 - y2) ** 2)
```

```
x1 = float(input())
```

```
x2 = float(input())
```

```
y1 = float(input())
```

```
y2 = float(input())
```

```
print(distance(x1, x2, y1, y2))
```

## Практическое занятие №11

```
# Выполнил Толоконников Алексей Михайлович П2-18
# 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 или большим числом пробелов или символами
# конца строки.
```

```
counter = {}
for word in input().split():
    counter[word] = counter.get(word, 0) + 1
print(counter[word] - 1, end=' ')
```

## Практическое занятие №12

```
# Выполнил Толоконников Алексей Михайлович П2-18.  
# Даны два списка чисел. Посчитайте, сколько чисел содержится  
# одновременно  
# как в первом списке, так и во втором.  
a={}  
a=set(input().split())  
b={}  
b=set(input().split())  
c=a.intersection(b)  
print(len(c))
```

## Практическое занятие №13

### Задача №1(кортеж)

```
# Выполнил Завадский Михаил П2-18.  
# Необходимо вывести все четные числа на отрезке [a; a * 10].  
a= int(input())  
b=a  
if b%2==1:  
    b=b+1  
print(tuple(range(b, a*10+1, 2)))
```

### Задача №2(кортеж)

```
# Выполнил Завадский Михаил П2-18.  
# С клавиатуры вводятся целые числа a > b.  
# Выведите убывающую последовательность чисел по одному числу в  
# строке.  
a=int(input())  
b=int(input())  
c=tuple(range(a,b,-1))  
for i in c:  
    print(i)
```

## Практическое занятие №14

```
# Выполнил Завадский Михаил Андреевич П2-18.  
# Дана строка S и текстовый файл. Добавить строку S в конец файла.  
v = open('text.txt', 'a')  
v.write('fwefwefwef')  
v.close()
```



## Практическое занятие №15

```
# Выполнил Толоконников Алексей Михайлович П2-18.
# Контейнерные типы данных модуля collections.
# Класс Counter() модуля collections в Python.

# 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
# >>> cnt = Counter(['red', 'blue', 'red', 'green', 'blue',
# 'blue'])
# >>> dict(cnt)
# {'blue': 3, 'red': 2, 'green': 1}

# Синтаксис:
# import collections
# cnt = collections.Counter([iterable-or-mapping])

# Параметры:
# iterable-or-mapping - итерируемая последовательность или
# словарь.

# Пример работы с ним:

import collections

c = ('g', 'b', 'c', 'd', 'g', 'f', 'f', 'f', 'g', 'c', 'd')
print('Оно помогает выводить количество неизменяемых
элементов.\nПоследовательность: ', c)
print('Сколько в ней элементов:')
print(collections.Counter(c))
```

## Практическое занятие №16

```
# Подготовлено Завадским Михаил Андреевичем П2-18
from tkinter import *
# импортируем библиотеку для работы с окнами
class Window:
# self говорит о том, что переменная принадлежит данному классу
    def __init__(self, width, height, title="MyProblem",
resizable=(False, False), icon=None):
# отвечает какими параметрами изначально владеет наше окно
        self.root = Tk()
# корневая переменная хранит Tk
        self.root.title(title)
# название нашего окна
        self.root.geometry(f"{width}x{height}+700+500")
# геометрия нашего окна (начальное расположение и разрешение)
        self.root.resizable(resizable[0], resizable[1])
# возможность изменения окна по координатам (x, y)
        if icon:
            self.root.iconbitmap(icon)
# иконка
        self.label = Label(self.root, text="Интересный текст")
# виджет label. Нужен для работы с текстом в окне
    def run(self):
        self.draw_widgets()
        self.root.mainloop()

    def draw_widgets(self):
        self.label.pack()
# настройка виджетов

# пользуемся классом Window :

from window import Window
# импортируем класс window
window = Window(400, 360)
# настраиваем параметры окна
window.run()
# запускаем окно
```