

# **ОТЧЕТ**

## **Лабораторная работа №1**

Работа со встроенными типами данных, числовые типы, строки,  
кортежи, изменяемые последовательности.

Применение основных арифметических операций, определение  
приоритетов

Выполнила: Балабанова Н.И.

Группа: ПИН-31Д

**Задание №1.** Пользователь с клавиатуры вводит произвольную строку. Необходимо вывести на экран YES, если все символы находятся в верхнем регистре и NO – в противном случае.

```
analyzedString = input("Введите строку:\n")

try:
    int(analyzedString)
    print("not string")
except ValueError:
    if analyzedString == "":
        print("empty string")
    elif analyzedString == analyzedString.upper():
        print("YES")
    else:
        print("NO")
```

Ввод	Вывод	Примечание
тест	NO	
Тест	NO	
ТЕСТ	YES	
1	not string	без проверки на число будет выводиться YES
ТЕСТ1	YES	число приравнивается к uppercase
тест1	NO	
	empty string	без проверки на пустоту будет выводиться YES

**Задание №2.** Пользователь с клавиатуры вводит имя и отчество, необходимо вывести на экран инициалы.

*Пример: Иван Сергеевич – И.С.*

```
import re

fio = input("Введите имя и отчество:\n")

separatedFio = fio.split(" ")

initials = ""

regexp = re.compile(r'[0-9]')

if len(separatedFio) != 2 or regexp.search(fio):
    print("wrong input")
else:
    for s in separatedFio:
        initials += s[0] + "."

    print(initials.upper())
```

Ввод	Вывод	Примечание
Иван Иванович	И.И.	
Иван Иванович Иванов	wrong input	
1 2 3	wrong input	
Иван	wrong input	
Иван 2	wrong input	
1 2	wrong input	
иван иванович	И.И.	
	wrong input	

**Задание №3.** Пользователь с клавиатуры вводит N целых чисел, необходимо вывести на экран только уникальные числа (не повторяющиеся).

```
analyzedString = input("Введите числа: \n")

arr = analyzedString.split(" ")

onlyNumbers = []

for n in arr:
    try:
        onlyNumbers.append(int(n))
    except ValueError:
        continue

setOfNumbers = set(onlyNumbers)

print("set:\t", setOfNumbers)
```

Ввод	Вывод	Примечание
1 2 3 4 5 1 2 3	1 2 3 4 5	
1	1	
1 2 3 r 1	1 2 3	char исключаются из анализируемого набора чисел
[]	{}	
1 2 3.5 6 3	1 2 3 6	float исключаются из анализируемого набора чисел

**Задание №4.** Напишите скрипт, который по введенному году выводит сообщение о том, является ли год високосным.

Для начала обозначим условия, по которым определяется високосный год, а также выделим тестовые данные:

- год, номер которого кратен 400, — високосный;
- остальные годы, номер которых кратен 100, — невисокосные (например, годы 1700, 1800, 1900, 2100, 2200, 2300);
- остальные годы, номер которых кратен 4, — високосные<sup>[5]</sup>;
- все остальные годы — невисокосные.

*Последние годы столетий, оканчивающиеся на два нуля, в трёх случаях из четырёх не являются високосными. Так, годы 1700, 1800 и 1900 не являются високосными, так как они кратны 100 и не кратны 400. Годы 1600 и 2000 — високосные, так как они кратны 400. Годы 2100, 2200 и 2300 — невисокосные.*

*Високосным годом был 2020 год, следующим будет 2024 год.*

```
def checkIfLeap(year):  
    if year % 4 == 0 and year % 100 != 0 or year % 4 == 0 and year % 400 == 0:  
        print("это високосный год")  
    else:  
        print("это обычный год")  
  
    return  
  
yearString = input("Введите год:\n")  
  
try:  
    checkIfLeap(int(yearString))  
except:  
    print("wrong input")
```

Ввод	Вывод	Ожидаемый вывод
2020	это високосный год	это високосный год
2021	это обычный год	это обычный год
1700	это обычный год	это обычный год
1600	это високосный год	это високосный год
2200	это обычный год	это обычный год
2024	это високосный год	это високосный год
2020y	wrong input	wrong input

**Задание №5.** Последовательно вводятся ненулевые числа. Определить сумму положительных и сумму отрицательных чисел. Закончить ввод чисел при вводе 0.

```
print("Введите ненулевые числа:\n")

positiveNumsSum = 0
negativeNumsSum = 0

while True:
    strNum = input()

    if strNum == '0':
        break

    try:
        num = float(strNum)

        if num > 0:
            positiveNumsSum += num
        else:
            negativeNumsSum += num
    except:
        print("not number, try again")

print("сумма положительных чисел:\t {0}\nсумма отрицательных чисел:\t{1}".format(positiveNumsSum, negativeNumsSum))
```

Ввод	Вывод	Примечание
1 -2 3 -4 5 -6	9.0 / -12.0	
1 2 r 4 5 6	18.0 / 0	ввод некорректных данных перехватывается, программа не завершается
0	0 / 0	
1.2 3.67 -4.80 -2.567	4.87 / -7.367	