**Лабораторная работа 5 Hello Python 2: структуры данных, функции и БД**

Цель: освоить простые приемы работы с различными структурами данных в

Python, научиться создавать и вызывать функции, познакомится с подходом

использования баз данных в Python.

**Задача 1: Палиндром**

Палиндром – это строка, которая читается одинаково слева направо и

справа налево.

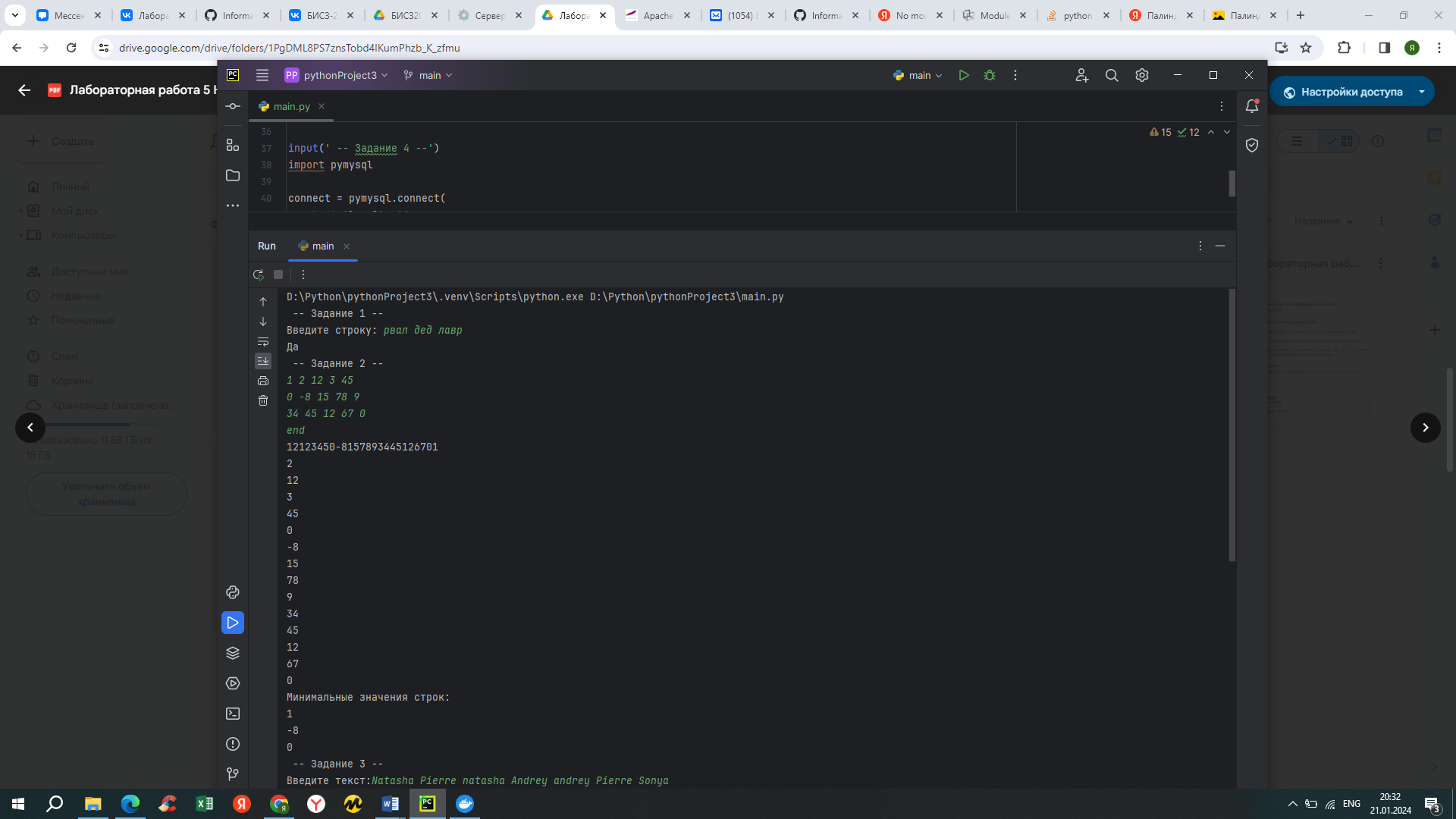
Напишите программу, которая получает строку, введенную с клавиатуры и,

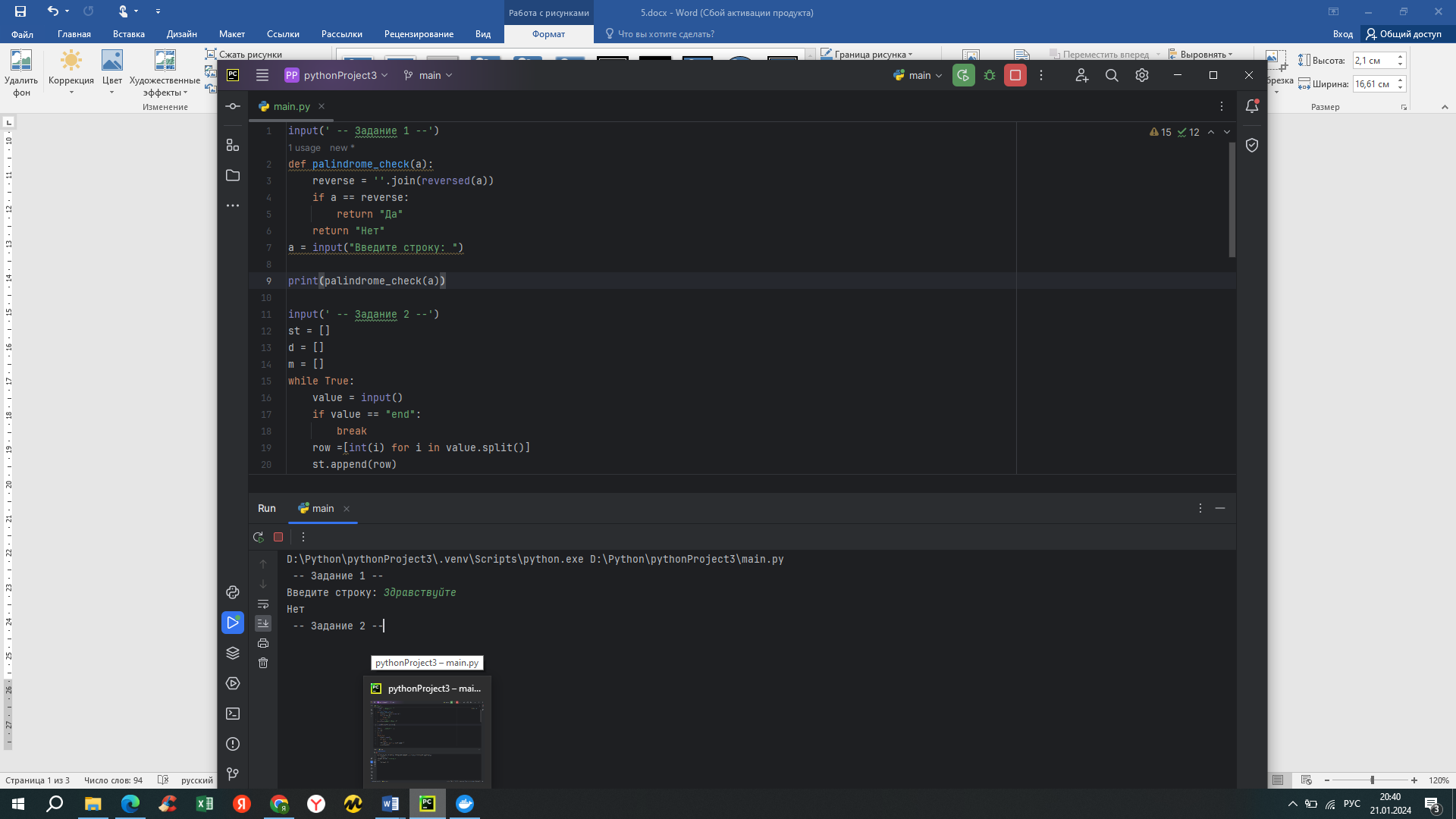
если введенная строка палиндром, выводит “Да”, иначе выводит “Нет”.

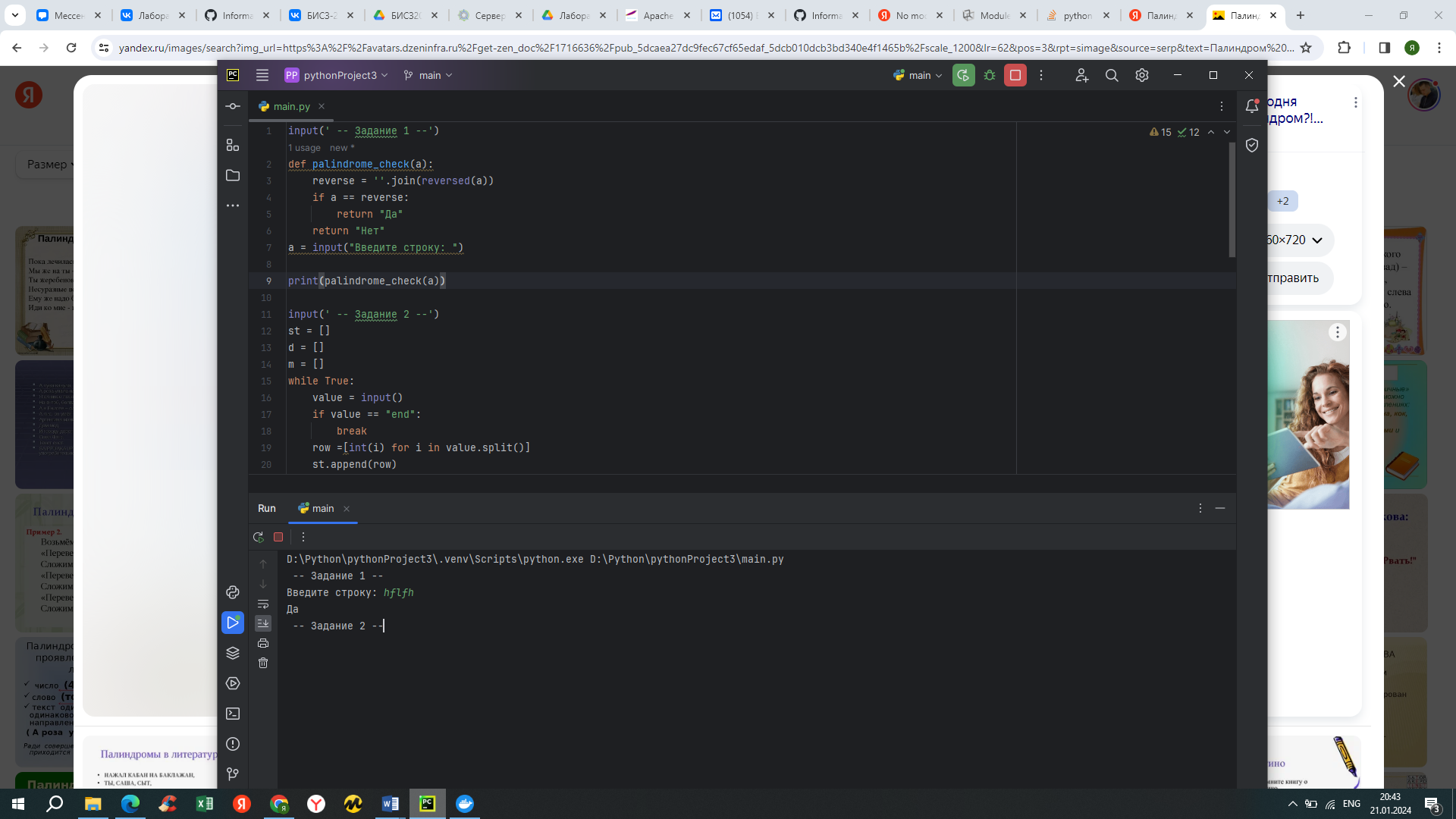
**Код:**

input(' -- Задание 1 --')  
def palindrome\_check(a):  
 reverse = ''.join(reversed(a))  
 if a == reverse:  
 return "Да"  
 return "Нет"  
a = input("Введите строку: ")  
  
print(palindrome\_check(a))

**Вывод:**







**Задача 2: Минимумы**

Напишите программу, на вход которой подаётся прямоугольная матрица в

виде последовательности строк (числа пишем через пробел). Используйте

метод split() строки. После последней строки матрицы идёт строка,

содержащая только end.

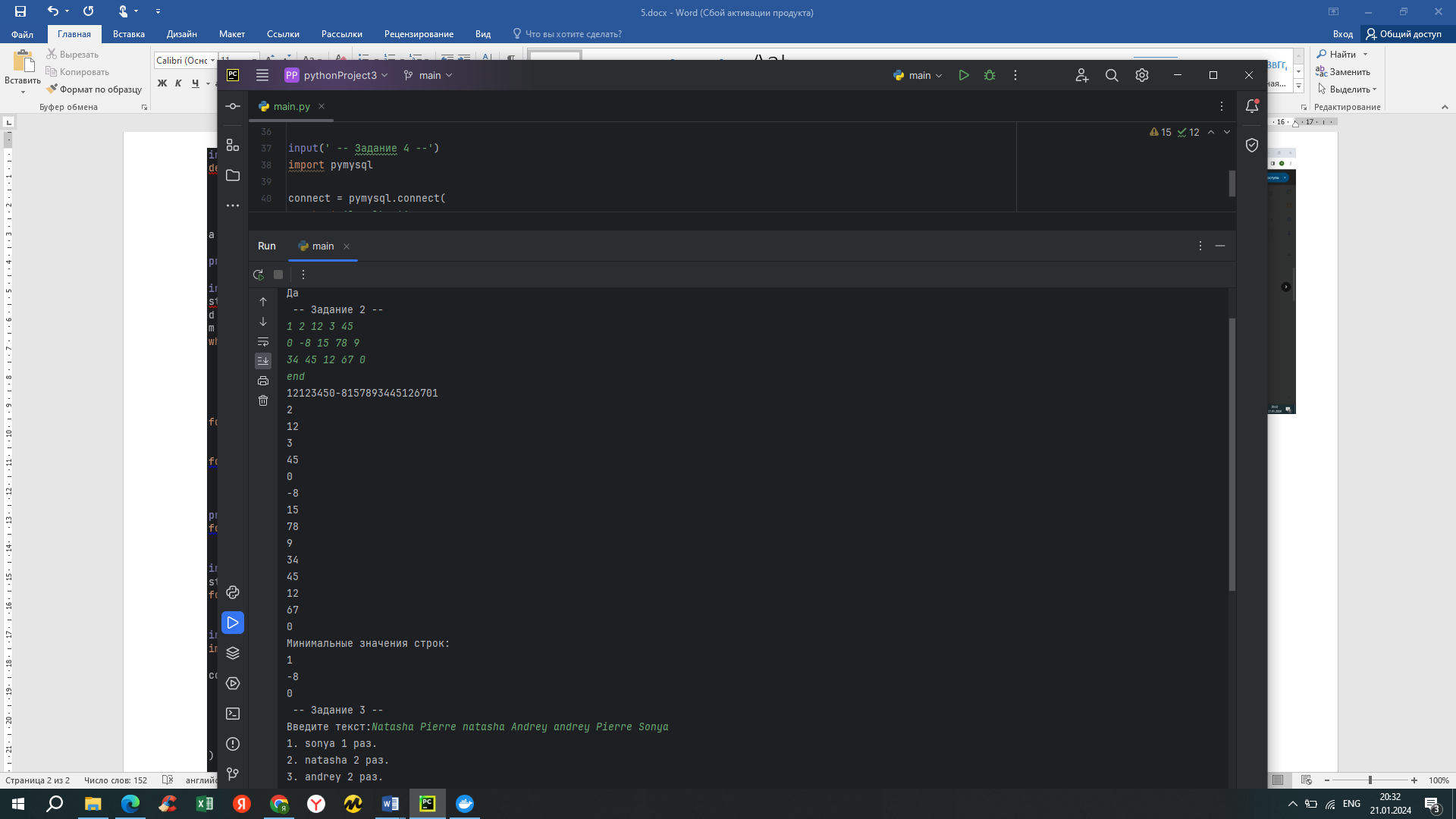
После того, как пользователь ввел end, программа должна найти в каждой

строке матрицы минимальное значение и вывести его на экран.

Код:

input(' -- Задание 2 --')  
st = []  
d = []  
m = []  
while True:  
 value = input()  
 if value == "end":  
 break  
 row =[int(i) for i in value.split()]  
 st.append(row)  
for i in range(len(st)):  
 for j in range(len(st[i])):  
 print(st[i][j], end = "", sep = "")  
for m in range(len(st)):  
 for n in range(len(st[0])):  
 print(st[m][n],end= '')  
 print()  
print("Минимальные значения строк:")  
for k in st:  
 print(min(k))

Вывод:



**Задача 3: Персонажи**

Во время чтения “Войны и мир” доктор Иванов заметил большое количество

персонажей и задался вопросом, а сколько персонажей всего и сколько раз

упоминается каждый персонаж. Напишите простой вариант программы, которая

подсчитывает сколько различных слов введено и сколько каждое слово

встречается.

Программа должна считывать одну строку со стандартного ввода и выводить

для каждого уникального слова в этой строке число его повторений (без учета

регистра) в формате "слово: количество".

Для работы программы нужно:

– использовать словарь (dict), где ключ – будет имя героя, а значение –

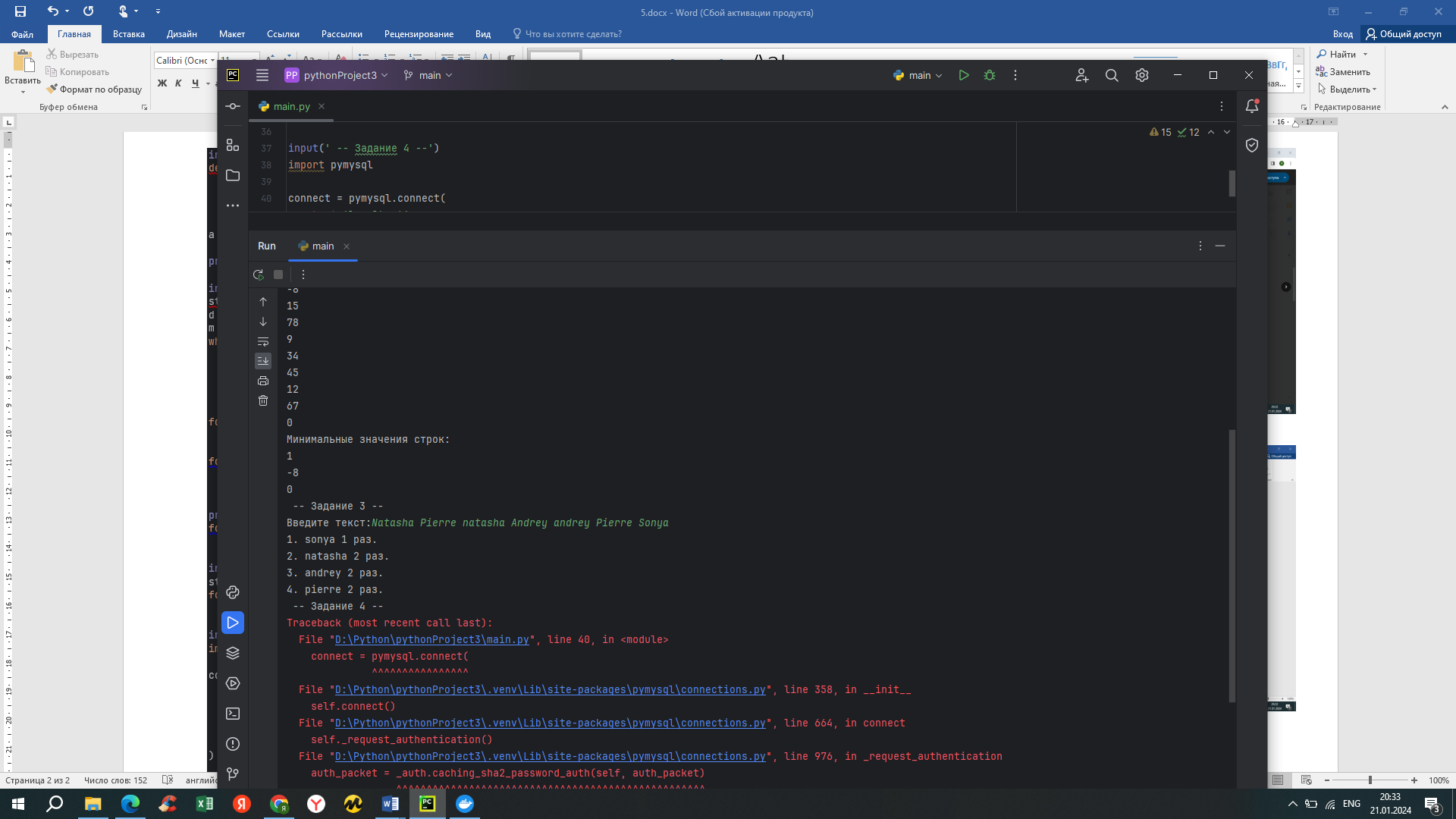
сколько раз встретилось имя.

– не забывать про последовательность методов строк s = input().lower().split()

**Код:**

input(' -- Задание 3 --')  
string = input('Введите текст:').lower().split()  
for i, word in enumerate(set(string), 1):  
 print(f'{i}. {word} {string.count(word)} раз.')

**Вывод:**



**Задача 4: MySQL**

Напишите программу, которая создает и отправляет запрос к БД MySQL, мы

уже создали в MySQL свою БД – my\_db.

Требования к программе:

– программа должна устанавливать соединение с СУБД MySQL, которая

запущена в Docker (контейнер с СУБД должен быть конечно запущен), для

установки соединения напишите отдельную функцию и сделайте её вызов в

начале программы;

– программа должна запрашивать введение года с клавиатуры;

– программа должна подставлять введенный год в запрос и отправлять

запрос в СУБД.

Требования к запросу:

– должен возвращать из таблицы user три поля: first\_name, last\_name и

date\_of\_birth;

– должен отбирать только тех пользователей, у которых год из даты

рождения равен заданному.

**Код:**

input(' -- Задание 4 --')  
import pymysql  
  
connect = pymysql.connect(  
 host='localhost',  
 port=3306,  
 user='root',  
 password='pass',  
 database='my\_db'  
)  
  
cur = connect.cursor()  
year= int(input('Введите год: '))  
s = year  
cur.execute(' select first\_name,last\_name,date\_of\_birth from user where year(date\_of\_birth) = %s',s)  
for rec in cur:  
 print(rec[0],rec[1],rec[2])

**Вывод:**

