

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Лабораторная работа 4**

Работа с исключениями в языке Python

Выполнил студент группы ИВТ-б-о-20-1

Пентухов С. А. « » \_\_\_\_\_ 2022г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 2022г.

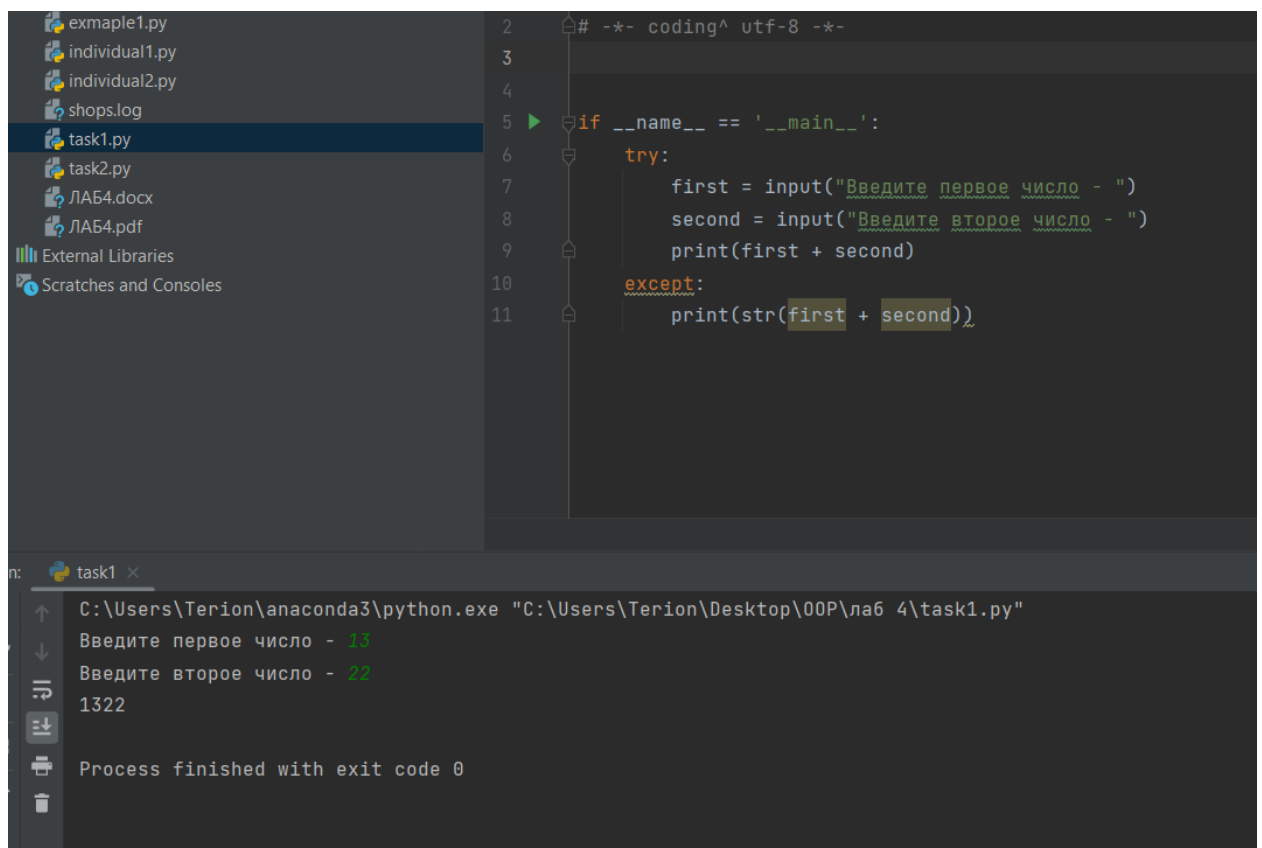
Проверил Воронкин Р.А. \_\_\_\_\_

(подпись)

**Цель работы:** приобретение навыков по перегрузке операторов при написании программ с помощью языка программирования Python версии 3.x.

## Задание 1

Решите следующую задачу: напишите программу, которая запрашивает ввод двух значений. Если хотя бы одно из них не является числом, то должна выполняться конкатенация, т. е. соединение, строк (рис. 1).



```
2  # -*- coding^ utf-8 -*-
3
4
5  if __name__ == '__main__':
6      try:
7          first = input("Введите первое число - ")
8          second = input("Введите второе число - ")
9          print(first + second)
10     except:
11         print(str(first + second))
```

task1

C:\Users\Terion\anaconda3\python.exe "C:\Users\Terion\Desktop\00P\лаб 4\task1.py"

Введите первое число - 13

Введите второе число - 22

1322

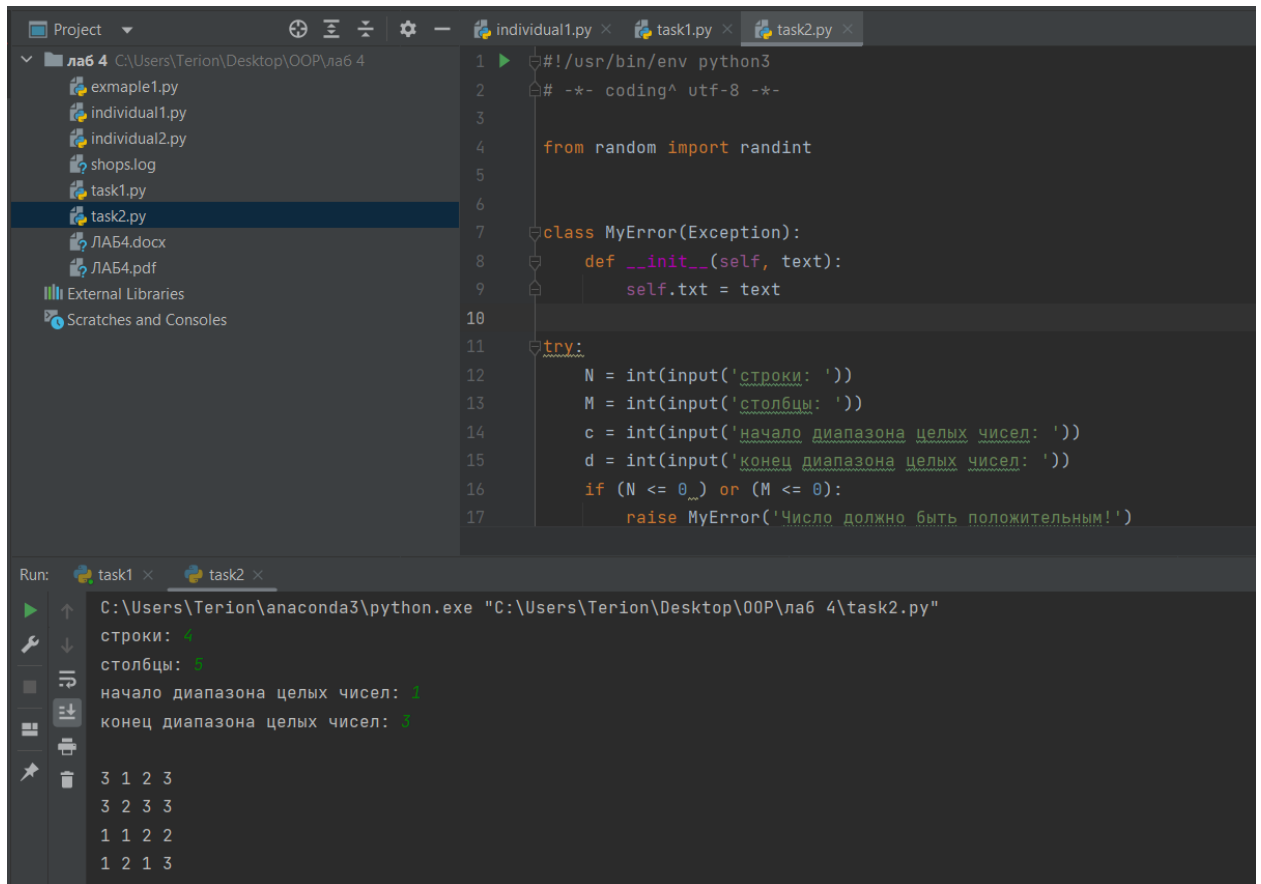
Process finished with exit code 0

Рисунок 1 – Результат выполнения при цифрах

## Задание 2

Решите следующую задачу: напишите программу, которая будет генерировать матрицу из случайных целых чисел. Пользователь может указать

число строк и столбцов, а также диапазон целых чисел. Произведите обработку ошибок ввода пользователя.



```
1 #!/usr/bin/env python3
2 # -*- coding^ utf-8 -*-
3
4 from random import randint
5
6
7 class MyError(Exception):
8     def __init__(self, text):
9         self.txt = text
10
11 try:
12     N = int(input('строки: '))
13     M = int(input('столбцы: '))
14     c = int(input('начало диапазона целых чисел: '))
15     d = int(input('конец диапазона целых чисел: '))
16     if (N <= 0) or (M <= 0):
17         raise MyError('Число должно быть положительным!')
```

Run: task1 × task2 ×

C:\Users\Terion\anaconda3\python.exe "C:\Users\Terion\Desktop\00P\лаб 4\task2.py"

строки: 4  
столбцы: 5  
начало диапазона целых чисел: 1  
конец диапазона целых чисел: 3

```
3 1 2 3
3 2 3 3
1 1 2 2
1 2 1 3
```

Рисунок 2 – Результат выполнения

## Индивидуальное задание 1

Выполнить индивидуальное задание 1 лабораторной работы 2.19, добавив возможность работы с исключениями и логгирование.

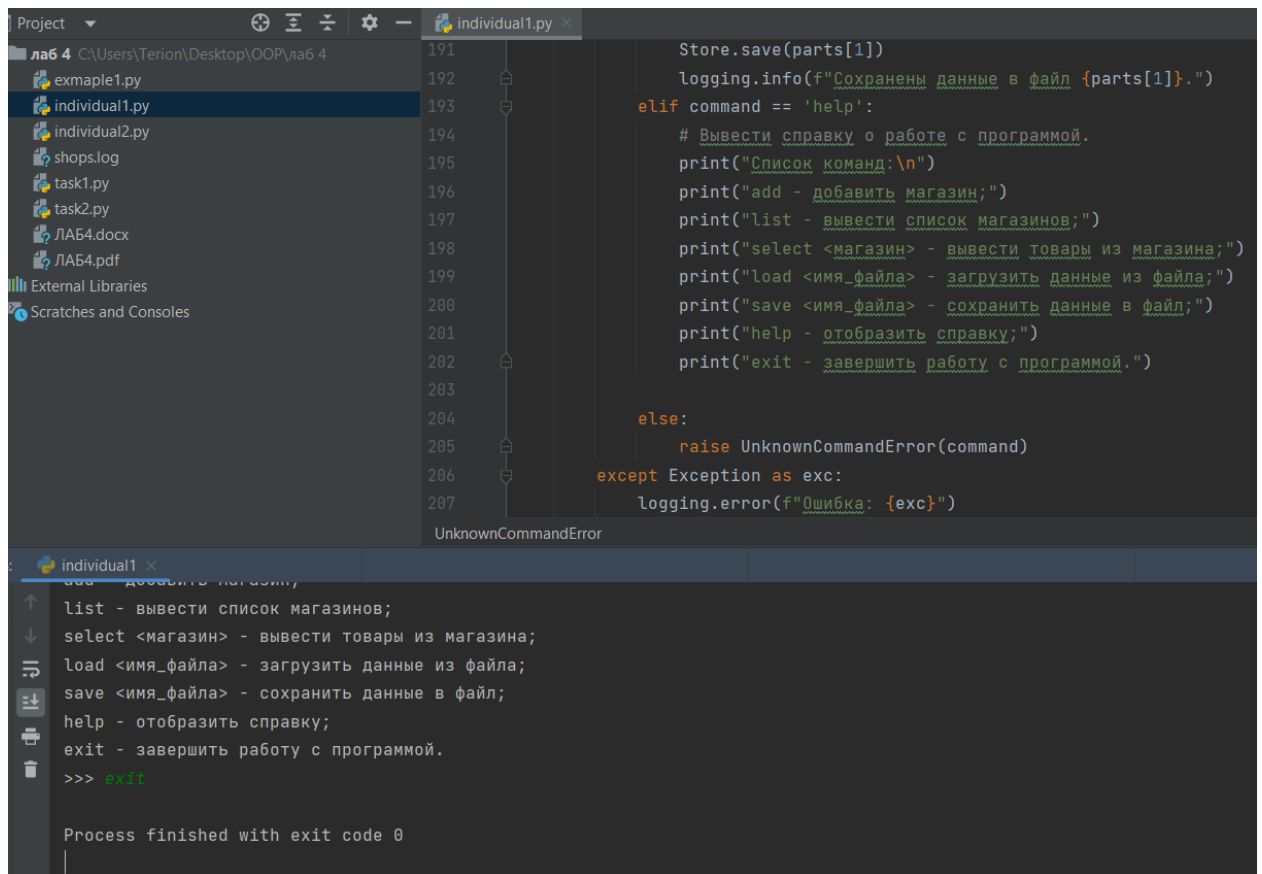


Рисунок 3 – Результат выполнения

## Индивидуальное задание 2

Изучить возможности модуля logging. Добавить для предыдущего задания вывод в файлы лога даты и времени выполнения пользовательской команды с точностью до миллисекунды.

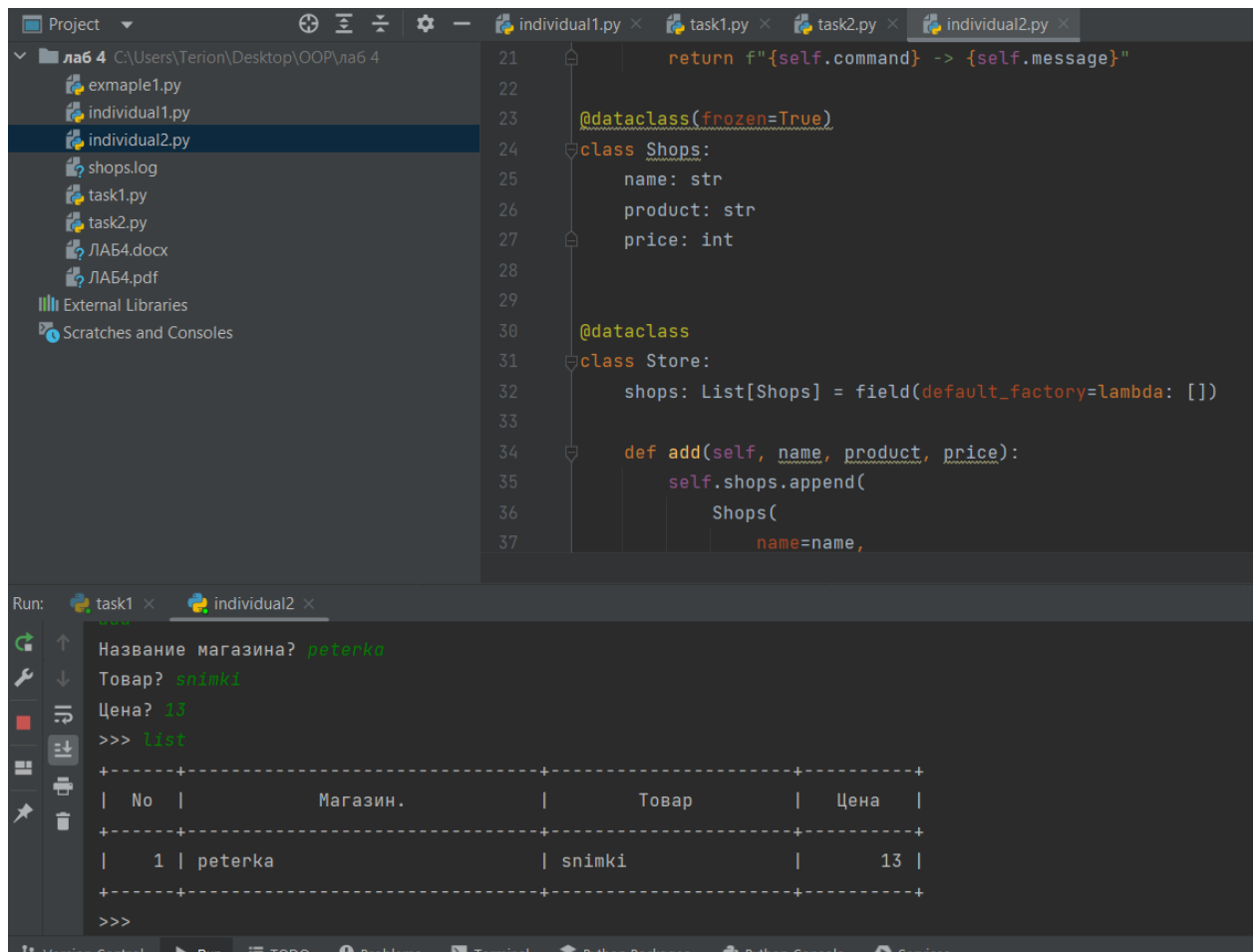


Рисунок 4 – Результат выполнения

## Ответы на контрольные вопросы

1. Как осуществляется объявление класса в языке Python? - Классы объявляются с помощью ключевого слова `class` и имени класса.
2. Чем атрибуты класса отличаются от атрибутов экземпляра? - Атрибуты класса определены внутри класса, но вне каких-либо методов. Их значения одинаковы для всех экземпляров этого класса. Так что вы можете рассматривать их как тип значений по умолчанию для всех наших объектов. Что касается переменных экземпляра, они хранят данные, уникальные для каждого объекта класса. В этой теме мы рассмотрим только атрибуты класса, но не волнуйтесь, у вас будет достаточно времени, чтобы узнать больше и об атрибутах экземпляра.
3. Каково назначение методов класса? - Методы определяют функциональность объектов, принадлежащих конкретному классу.
4. Для чего предназначен метод `__init__()` класса? – Для объявления конструктора класса.
5. Каково назначение `self` ? – Указание объекта на самого себя.
6. Как добавить атрибуты в класс? - Атрибуты экземпляра - это как раз те, которые мы определяем в методах, поэтому по определению мы можем создавать новые атрибуты внутри наших пользовательских методов.

Как осуществляется управление доступом к методам и атрибутам в языке Python? - т.д.) - В Python таких возможностей нет, и любой может обратиться к атрибутам и методам вашего класса, если возникнет такая необходимость. Это существенный недостаток этого языка, т.к. нарушается один из ключевых принципов ООП – инкапсуляция. Хорошим тоном считается, что для чтения/изменения какого-то атрибута должны использоваться специальные методы, которые называются `getter/setter`, их можно реализовать, но ничего не мешает изменить атрибут напрямую. При этом есть соглашение, что метод или атрибут, который начинается с нижнего подчеркивания, является скрытым, и снаружи класса трогать его

не нужно (хотя сделать это можно).

7. В Python есть встроенная функция `instance ()`, которая сравнивает значение с указанным типом. Если данное значение и тип соответствуют, он вернет `true`, иначе `false`. Используя `isinstance ()`, вы можете проверить строку, число с плавающей точкой, `int`, список, кортеж, `dict`, `set`, `class` и т. Д.