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

Кафедра инфокоммуникаций

Отчет по лабораторной работе №1

Элементы объектно-ориентированного программирования в языке Python

по дисциплине «Объектно-ориентированное программирование»

Выполнил студент группы ИВТ-б-о-20-1
Павленко М.С. « »_____20__г.
Подпись студента_____
Работа защищена« »_____20__г.

Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2023

Цель работы: приобретение навыков по работе с классами и объектами при написании программ с помощью языка программирования Python версии 3.x. Ход работы

1. Изучив методические указания, приступил к разбору примера.

```
3/4
Введите обыкновенную дробь: 5/6
5/6
19/12
1/12
5/8
10/9
```

Рисунок 1.1 – Проверка правильности работы кода

2. Затем начал выполнять задания для моего варианта.

```
const_list = [1, 2, 5, 10, 50, 100, 500, 1000, 5000]
def __init__(self):
    self.sum_dict = {}
    self.summ = 0
    self.first = 0
    self.second = 0

1 usage
def read(self, prompt=None):
    self.first = input('Введите номинал купюры: ') if prompt is None else input(prompt)
    if int(self.first) in TaskOne.const_list:
        self.second = input('Введите количество купюр данного номинала: ') if prompt is None else input(prompt)
        self.sum_dict[self.first] = self.second
    else:
        print("Такого номинала нет в списке")

1 usage
def display(self):
    print("Сумма: ", make_summa(self.sum_dict, self.summ))


1 usage
def make_summa(sum_dict, summ):
    with open('new_json.json', 'r', encoding='utf-8') as file:
        json_list = json.load(file)
    json_list.append(sum_dict)
    for i in json_list:
        for k, j in i.items():
            summ += int(k) * int(j)
    with open('new_json.json', 'w', encoding='utf-8') as file:
        json.dump(json_list, file)
```

Рисунок 1.2 – Код первого индивидуального задания

```
(base) C:\Users\zligo\Documents\GitHub\проекты\ООП\Лаба 1>python "Задание 1".py
Список доступных номиналов: 1, 2, 5, 10, 50, 100, 500, 1000, 5000
Введите номинал купюры: 1000
Введите количество купюр данного номинала: 2
Сумма:  2000
Введите номинал купюры: 51
Такого номинала нет в списке
Сумма:  2000
Введите номинал купюры: 5
Введите количество купюр данного номинала: 1
Сумма:  2005
Введите номинал купюры: 10
Введите количество купюр данного номинала: 2
Сумма:  2025
Введите номинал купюры:
```

Рисунок 1.3 – Проверка кода первого задания

```
            print("Треугольник прямоугольный")
        elif self.a == self.b or self.b == self.c or self.a == self.c:
            print("Треугольник равнобедренный")


if __name__ == "__main__":
    command_line = None
    Tri = Triangle()
    file_parser = argparse.ArgumentParser(add_help=False)
    parser = argparse.ArgumentParser("flights")
    parser.add_argument(
        "--version",
        action="version",
        version="%(prog)s 0.1.0")
    subparsers = parser.add_subparsers(dest="command")
    read = subparsers.add_parser(
        "read",
        parents=[file_parser])
    square = subparsers.add_parser(
        "square",
        parents=[file_parser])
    perimeter = subparsers.add_parser(
        "perimeter",
        parents=[file_parser])
    heights = subparsers.add_parser(
        "heights",
        parents=[file_parser])
    view = subparsers.add_parser(
        "view",
        parents=[file_parser])
    display = subparsers.add_parser(
```

Рис. 2 – Код второго задания

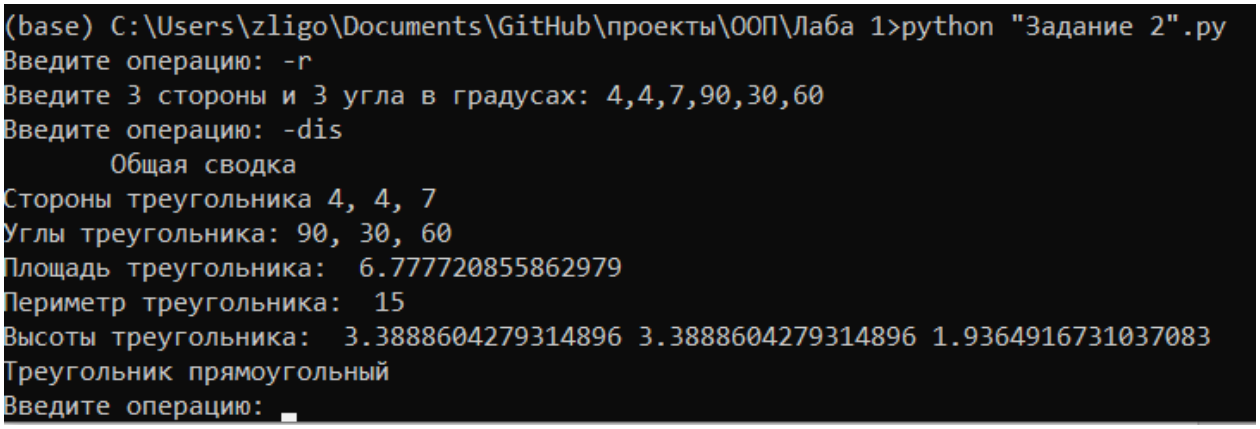

Рисунок 2.1 – Проверка кода второго заданияКонтрольные вопросы

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.

3. Каково назначение методов класса?

Методы определяют функциональность объектов, принадлежащих конкретному классу.

4. Для чего предназначен метод___init__() класса?

Метод____init___позволяет принимать аргументы для вашего класса. Чтоеще более важно, метод__init____дает возможность назначать начальные значения различным атрибутам экземпляров класса.

5. Каково назначение self ?

Self - это обращение к самому экземпляру класса.

6. Как добавить атрибуты в класс?

“Имя объекта”.”Название атрибута” = “Значение атрибута”

7. Как осуществляется управление доступом к методам и атрибутам в языке Python?

Для чтения/изменения какого-то атрибута должны использоваться специальные методы, которые называются getter/setter.

8. Каково назначение функции isinstance ?

Функция isinstance() вернет True, если проверяемый объект object является экземпляром указанного класса (классов) или его подкласса (прямого, косвенного или виртуального).

Если объект object не является экземпляром данного типа, то функция всегда возвращает False.

Вывод: в ходе выполнения лабораторной работы были приобретены простейшие навыки по работе с классами, экземплярами, методами и свойствами в языке программирования Python.