

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

Кафедра инфокоммуникаций

Отчет по лабораторной работе №4.1

**Элементы объектно-ориентированного программирования в языке
Python**

По дисциплине «Объектно-ориентированное программирование»

Выполнил студент группы ИВТ-б-о-20-1

Плотников Д. В. « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

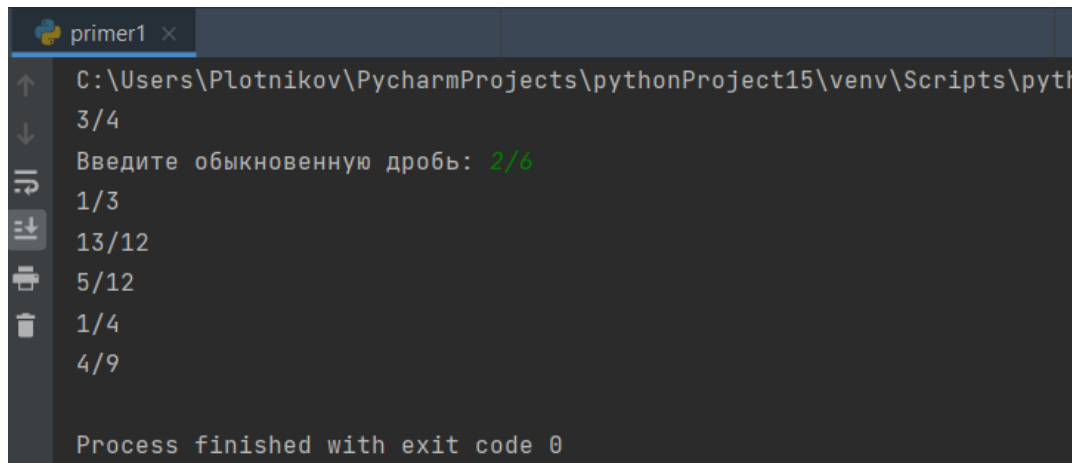
Проверил Воронкин Р. А. _____

(подпись)

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

Ход работы

1. Создал новый собственный репозиторий. Ссылка на репозиторий: https://github.com/Dmitry-15/4.1_laba.
2. С помощью команды `git clone` клонировал удаленный репозиторий на свой ПК. Дополнил файл `.gitignore` необходимыми правилами для работы с IDE PyCharm.
3. Далее проработал пример.



```
primer1 x
C:\Users\Plotnikov\PycharmProjects\pythonProject15\venv\Scripts\python.exe
3/4
Введите обыкновенную дробь: 2/6
1/3
13/12
5/12
1/4
4/9
Process finished with exit code 0
```

Рисунок 1 – Выполнение примера

Индивидуальные задания

Вариант 14

Задание 1

1. Условие задачи: поле `first` — дробное положительное число, оклад; поле `second` — целое число, количество отработанных дней в месяце. Реализовать метод `summa ()` — вычисление начисленной суммы за данное количество дней для заданного месяца: $\text{оклад} / \text{дни месяца} * \text{отработанные дни}$.

```
individ1 x
C:\Users\Plotnikov\PycharmProjects\pythonProject15\venv\Scripts\python.exe
Начисленная сумма за данное количество дней: 23750.416666666668
Введите дробное число, оклад: 31211.65
Введите целое число, число отработанных дней в месяце: 28
Введите целое число, количество дней в месяце: 30
Начисленная сумма за данное количество дней: 29130.873333333333
Process finished with exit code 0
```

Рисунок 2 – Выполнение первого индивид. задания

Задание 2

1. Условие задачи: создать класс Payment (зарплата). В классе должны быть представлены поля: фамилия-имя-отчество, оклад, год поступления на работу, процент надбавки, подоходный налог, количество отработанных дней в месяце, количество рабочих дней в месяце, начисленная и удержанная суммы. Реализовать методы: вычисления начисленной суммы, вычисления удержанной суммы, вычисления суммы, выдаваемой на руки, вычисления стажа. Стаж вычисляется как полное количество лет, прошедших от года поступления на работу, до текущего года. Начисления представляют собой сумму, начисленную за отработанные дни, и надбавки, то есть доли от первой суммы. Удержания представляют собой отчисления в пенсионный фонд (1% от начисленной суммы) и подоходный налог. Подоходный налог составляет 13% от начисленной суммы без отчислений в пенсионный фонд.

```
individ2 x
C:\Users\Plotnikov\PycharmProjects\pythonProject15\venv\Scripts\python.exe
Начисленная сумма: 28750 руб.
Удержанная сумма: 3500 руб.
Сумма выданная на руки: 25250 руб.
Трудовой стаж: 12 лет
Введите ФИО: Плотников Дмитрий Владимирович
Введите оклад: 50000
Введите год вашего поступления на работу: 2010
Введите процент надбавки: 15
Введите количество отработанных дней в месяце: 29
Введите количество рабочих дней в месяце: 30
Начисленная сумма: 55583 руб.
Удержанная сумма: 6767 руб.
Сумма выданная на руки: 48817 руб.
Трудовой стаж: 4 лет
Process finished with exit code 0
```

Рисунок 3 – Выполнение второго индивид. задания

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.

3. Каково назначение методов класса?

Методы определяют функциональность объектов, принадлежащих конкретному классу.

4. Для чего предназначен метод `__init__()` класса?

Метод `__init__` является конструктором. Конструкторы - это концепция объектно-ориентированного программирования. Класс может иметь один и только один конструктор. Если `__init__` определен внутри класса, он автоматически вызывается при создании нового экземпляра класса.

5. Каково назначение `self`?

Аргумент `self` представляет конкретный экземпляр класса и позволяет нам получить доступ к его атрибутам и методам. В примере с `__init__` мы создаем атрибуты для конкретного экземпляра и присваиваем им значения аргументов метода. Важно использовать параметр `self` внутри метода, если мы хотим сохранить значения экземпляра для последующего использования.

6. Как добавить атрибуты в класс?

Атрибут класса – это атрибут, общий для всех экземпляров класса. Атрибуты экземпляра – это как раз те, которые мы определяем в методах, поэтому по определению мы можем создавать новые атрибуты внутри наших пользовательских методов.

7. Как осуществляется управление доступом к методам и атрибутам

языке Python?

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).

8. Каково назначение функции `isinstance`?

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.

Вывод: в ходе выполнения лабораторной работы приобрел навыки по работе с классами и объектами при написании программ с помощью языка программирования Python версии 3.x.