

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

Кафедра инфокоммуникаций

Отчет по лабораторной работе №4.3

Наследование и полиморфизм 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 созданию иерархии классов 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-15/4.3_laba.
2. Ознакомившись с теорией выполнил для начала примеры.

```
C:\tools\Anaconda3\python.exe C:/Users/Plotnikov/PycharmProjects/4.3_laba/primer1.py
3/4
Введите обыкновенную дробь: 5/6
5/6
19/12
1/12
5/8
10/9
```

Рисунок 1 – Выполнение первого примера

```
C:\tools\Anaconda3\python.exe C:/Users/Plotnikov/PycharmProjects/4.3_laba/primer2.py
I have 3 sides
I have 4 sides
I have 5 sides
I have 6 sides

Process finished with exit code 0
```

Рисунок 2 – Выполнение второго примера

```
C:\tools\Anaconda3\python.exe C:/Users/Plotnikov/PycharmProjects/4.3_laba/primer3.py
I can walk and run
I can crawl
I can bark
I can roar

Process finished with exit code 0
```

Рисунок 3 – Выполнение третьего примера

3. Приступил к выполнению индивидуальных заданий.

Индивидуальные задания

Вариант 18

Задание 1

1. Условие задания: создать класс Pair (пара чисел); определить метод перемножения полей и операцию сложения пар $(a,b) + (c,d) = (a + b, c + d)$.

Определить производный класс Long с полями: старшая часть числа и младшая часть числа. Определить методы умножения и вычитания.

```
C:\tools\Anaconda3\python.exe C:/Users/Plotnikov/4.3_laba/Tasks/individ1.py
193310605386671245944316 15212422990198885227084
1064713914208 251498889656
-3982 2334
-31 -68

Process finished with exit code 0
```

Рисунок 4 – Выполнение первого индивид. задания

Задание 2

1. Условие задания: Создать абстрактный базовый класс Figure с абстрактными методами вычисления площади и периметра. Создать производные классы: Rectangle (прямоугольник), Circle (круг), Trapezium (трапеция) со своими функциями площади и периметра. Самостоятельно определить, какие поля необходимы, какие из них можно задать в базовом классе, а какие – в производных.

```
C:\tools\Anaconda3\python.exe C:/Users/Plotnikov/4.3_laba/Tasks/individ2.py
Площадь прямоугольника: 80
Периметр прямоугольника: 18
Площадь треугольника: 9.0
Периметр треугольника: 18
Площадь трапеции: 16.0
Периметр трапеции: 19

Process finished with exit code 0
```

Рисунок 5 – Выполнение второго индивид. задания

Контрольные вопросы:

1. Что такое наследование как оно реализовано в языке Python?

Наследование — это возможность расширения (наследования) ранее написанного программного кода класса с целью дополнения, усовершенствования или привязки под новые требования.

Синтаксически создание класса с указанием его родителя выглядит так:
class имя_класса(имя_родителя1, [имя_родителя2,..., имя_родителя_n])

2. Что такое полиморфизм и как он реализован в языке Python?

Полиморфизм - это способность выполнять действие над объектом независимо от его типа. Это обычно реализуется путем создания базового класса и наличия двух или более подклассов, которые все реализуют методы с одинаковой сигнатурой.

3. Что такое "утиная" типизация в языке программирования Python?

Эта концепция адаптирована из следующего абдуктивного умозаключения:

Если что-то выглядит как утка, плавает как утка и крикает как утка, это наверняка и есть утка.

Концепция утиной типизации в основном принята в языках программирования, поддерживающих динамическую типизацию, таких как Python и JavaScript. Общей особенностью этих языков является возможность объявления переменных без указания их типа.

При использовании пользовательских типов для определённых целей, реализация связанных функций важнее, чем точные типы данных.

Утиная типизация подчёркивает реализацию связанных выполняемых функций, а конкретные типы данных менее важны

4. Каково назначение модуля abc языка программирования Python?

Начиная с версии языка 2.6 в стандартную библиотеку включается модуль abc, добавляющий в язык абстрактные базовые классы (далее АБК).

АБК позволяют определить класс, указав при этом, какие методы или свойства обязательно переопределить в классах-наследниках.

5. Как сделать некоторый метод класса абстрактным?

Перед методом класса необходимо добавить декоратор модуля abc: @abstractmethod.

6. Как сделать некоторое свойство класса абстрактным?

Абстрактные классы включают в себя атрибуты в дополнение к методам, вы можете потребовать атрибуты в конкретных классах, определив их с помощью @abstractproperty.

7. Каково назначение функции isinstance ?

Функция `isinstance ()` в Python используется для проверки, является ли объект экземпляром указанного класса или нет.

Вывод: в ходе выполнения лабораторной работы успешно были приобретены простейшие навыки по работе с наследованием и абстрактными методами классов в языке программирования Python.