**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «ПиКЯП (Парадигмы и конструкции языков программирования)»

Отчет по лабораторной работе №2

«**Объектно-ориентированные возможности языка Python**»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| Студент группы ИУ5-31Б |  | Преподаватель каф. ИУ5 |
| Нагдасёв Д.М. |  | Гапанюк Ю.Е. |
|  |  |  |

Москва, 2024 г

**Цель лабораторной работы:** изучение объектно-ориентированных возможностей языка Python.

Код программы:

1. Файл **main.py**

from GeometricShape import \*

from ShapeColor import \*

from Rectangle import \*

from Circle import \*

from Square import \*

if \_\_name\_\_ == "\_\_main\_\_":

    N = 8

    rect = Rectangle(N, N + 2, 'синий')

    circ = Circle(N, 'зеленый')

    sq = Square(N, 'красный')

    print(rect)

    print(circ)

    print(sq)

2. Файл **Circle.py**

import math

from GeometricShape import \*

from ShapeColor import \*

class Circle(GeometricShape):

    shape\_name = "Круг"

    def \_\_init\_\_(self, radius, color):

        self.radius = radius

        self.color = ShapeColor(color)

    def area(self):

        return math.pi \* self.radius \*\* 2

    def \_\_repr\_\_(self):

        return "Название: {}, Радиус: {}, Цвет: {}, Площадь: {:.2f}".format(

            self.name(), self.radius, self.color.color, self.area()

        )

    def name(cls):

        return cls.shape\_name

3. Файл **GeometricShape.py**

from abc import ABC, abstractmethod

class GeometricShape(ABC):

    @abstractmethod

    def area(self):

        pass

    @abstractmethod

    def \_\_repr\_\_(self):

        pass

    @abstractmethod

    def name(cls):

        pass

4. Файл **Rectangle.py**

from GeometricShape import \*

from ShapeColor import \*

class Rectangle(GeometricShape):

    shape\_name = "Прямоугольник"

    def \_\_init\_\_(self, width, height, color):

        self.width = width

        self.height = height

        self.color = ShapeColor(color)

    def area(self):

        return self.width \* self.height

    def \_\_repr\_\_(self):

        return "Название: {}, Ширина: {}, Высота: {}, Цвет: {}, Площадь: {}".format(

            self.name(), self.width, self.height, self.color.color, self.area()

        )

    def name(cls):

        return cls.shape\_name

5. Файл **ShapeColor.py**

class ShapeColor:

    def \_\_init\_\_(self, color):

        self.color = color

6. Файл **Square.py**

from Rectangle import \*

class Square(Rectangle):

    shape\_name = "Квадрат"

    def \_\_init\_\_(self, side, color):

        super().\_\_init\_\_(side, side, color)

    def name(cls):

        return cls.shape\_name

**Результаты выполнения программы**

