



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

Факультет «ГУИМЦ»

Кафедра ИУ5 «Системы обработки информации и управления»

Дисциплина «Базовые компоненты ИТ»

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

«Объектно-ориентированные возможности языка Python»

Студент: Амосов П.А., группа ИУ5Ц-52Б

Преподаватель: Канев А.И.

2022г.

1. Описание задания

1) Необходимо создать виртуальное окружение и установить в него хотя бы один внешний пакет с использованием `pip`.

2) Необходимо разработать программу, реализующую работу с классами.

3) Все файлы проекта (кроме основного файла `main.py`) должны располагаться в пакете `lab_python_oop`.

4) Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета `lab_python_oop`.

5) Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры.

6) Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.

7) Класс «Прямоугольник» наследуется от класса «Геометрическая фигура». Класс должен содержать конструктор по параметрам «ширина», «высота» и «цвет». В конструкторе создается объект класса «Цвет фигуры» для хранения цвета. Класс должен переопределять метод, вычисляющий площадь фигуры.

8) Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа `math.pi` из модуля `math`.

9) Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны. Для классов «Прямоугольник», «Квадрат», «Круг»:

- Определите метод `"repr"`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод `format` - <https://pyformat.info/>

- Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.

12) В корневом каталоге проекта создайте файл `main.py` для тестирования Ваших классов (используйте следующую конструкцию - https://docs.python.org/3/library/__main__.html). Создайте следующие объекты и выведите о них информацию в консоль (N - номер Вашего варианта по списку группы):

Прямоугольник синего цвета шириной N и высотой N.

Круг зеленого цвета радиусом N.

Квадрат красного цвета со стороной N.

Также вызовите один из методов внешнего пакета, установленного с использованием pip.

2. Текст программы

main.py

```
from lab_python_oop.box import *
from lab_python_oop.rectangle import *
from lab_python_oop.circle import *
import numpy as np

def main():
    r = Rectangle("синего", 1, 1)
    c = Circle("зеленого", 1)
    s = Box("красного", 1)
    print(r)
    print(c)
    print(s)

    mas = np.array([[1,5,6,5],
                    [3,9,-2,0],
                    [9,4,7,1],
                    [1,0,9,4]])

    print(f"\n{np.linalg.det(mas)}")
    print(mas.dot(mas))

if __name__ == '__main__':
    main()
```

rectangle.py

```
from lab_python_oop.geometric_shape import Geometric_shape
from lab_python_oop.color import Color

class Rectangle(Geometric_shape):
    """
    Класс «Прямоугольник» наследуется от класса «Геометрическая фигура».
    """
    FIGURE_TYPE = "Прямоугольник"
    def __init__(self, color, width, height):
        self.width = width
        self.height = height
        self.color = Color()
        self.color = color

    def square(self):
        return self.width * self.height

    @classmethod
    def get_figure_type(cls):
        return cls.FIGURE_TYPE

    def __repr__(self):
```

```

        return '{} {} цвета шириной {} и высотой {} площадью {}'.format(
            Rectangle.get_figure_type(),
            self.color,
            self.width,
            self.height,
            self.square()
        )

```

circle.py

```

from lab_python_oop.geometric_shape import Geometric_shape
from lab_python_oop.color import Color
from math import pi

class Circle(Geometric_shape):
    """
    Класс «Круг» наследуется от класса «Геометрическая фигура».
    """
    FIGURE_TYPE = "Круг"
    def __init__(self, color, radius):
        self.radius = radius
        self.color = Color()
        self.color = color

    def square(self):
        return pi * self.radius ** 2

    @classmethod
    def get_figure_type(cls):
        return cls.FIGURE_TYPE

    def __repr__(self):
        return '{} {} цвета радиуса {} и площадью {}'.format(
            Circle.get_figure_type(),
            self.color,
            self.radius,
            self.square()
        )

```

color.py

```

class Color:
    """
    Класс «Цвет фигуры»
    """
    def __init__(self):
        self._color = None

    @property
    def color(self):
        return self._color

    @color.setter
    def color(self, value):
        self._color = value

    @color.deleter
    def color(self):
        del self._color

```

box.py

```
from lab_python_oop.rectangle import Rectangle

class Box(Rectangle):
    """
    Класс «Квадрат» наследуется от класса «Прямоугольник».
    """
    FIGURE_TYPE = "Квадрат"
    def __init__(self, color, width):
        super().__init__(color, width, width)

    @classmethod
    def get_figure_type(cls):
        return cls.FIGURE_TYPE

    def __repr__(self):
        return '{} {} цвета длиной {} площадью {}'.format(
            Box.get_figure_type(),
            self.color,
            self.width,
            self.square()
        )
```

geometric_shape.py

```
from abc import ABC, abstractmethod

class Geometric_shape(ABC):
    """
    Класс «Геометрическая фигура».
    """
    FIGURE_TYPE = "Геометрическая фигура"

    @abstractmethod
    def square(self):
        pass
```

3. Экранные формы с примерами выполнения программы

Прямоугольник синего цвета шириной 1 и высотой 1 площадью 1.

Круг зеленого цвета радиуса 1 и площадью 3.141592653589793.

Квадрат красного цвета длиной 1 площадью 1.

716.00000000000007

[[75 74 83 31]

[12 88 -14 13]

[85 109 104 56]

[86 41 105 30]]

Process finished with exit code 0