

**Московский государственный технический университет
им. Н. Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Разработка интернет-приложений»

Отчет по лабораторной работе №2

Вариант 17.

Выполнил:

Студент группы ИУ5-51Б

Ноздрова Валентина

Проверил:

Гапанюк Ю. Е.

Дата: 10.10.2021

Дата:

Подпись:

Подпись:

Москва, 2021 г.

Задание:

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. Программа должна быть разработана в виде консольного приложения на языке Python 3.
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. Подробнее про абстрактные классы и методы Вы можете прочитать [здесь](#).
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать [здесь](#).
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`
 - Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
10. В корневом каталоге проекта создайте файл `main.py` для тестирования Ваших классов. 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.

Листинг программы:

```
-----figure.py-----
import abc
from abc import ABC, abstractmethod

# Абстрактный класс "Геометрическая фигура"
class Figure(ABC):
    @abc.abstractmethod
    def square(self):
        pass

-----color.py-----
class FColor:

    def __init__(self):
        self._color = None

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

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

-----rectangle.py-----

from lab_python_oop.figure import Figure
from lab_python_oop.color import FColor

class Rectangle(Figure):

    _type = "прямоугольник"

    @classmethod
    def gettype(cls):
        return cls._type

    def __init__(self, p_width, p_height, p_color):
        self._width = p_width
        self._height = p_height
        self._color = FColor()
        self._color.colorprop = p_color

    def square(self):
        return self._width * self._height

    def __repr__(self):
        return "{} {} шириной {}, высотой {} и площадью {}".format(
            self._color.colorprop,
            Rectangle.gettype(),
            self._width,
            self._height,
            self.square()
        )
```

-----circle.py-----

```
from lab_python_oop.figure import Figure
from lab_python_oop.color import FColor
import math

class Circle(Figure):

    _type = "круг"

    @classmethod
    def gettype(cls):
        return cls._type

    def __init__(self, p_radius, p_color):
        self._radius = p_radius
        self._color = FColor()
        self._color.colorprop = p_color

    def square(self):
        return self._radius**2 * math.pi

    def __repr__(self):
        return "{} {} радиусом {} и площадью {}".format(
            self._color.colorprop,
            Circle.gettype(),
            self._radius,
            self.square()
        )
```

-----square.py-----

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.color import FColor

class Square(Rectangle):

    _type = "квадрат"

    @classmethod
    def gettype(cls):
        return cls._type

    def __init__(self, p_side, p_color):
        self._side = p_side
        self._color = FColor()
        self._color.colorprop = p_color

    def square(self):
        return self._side**2

    def __repr__(self):
        return "{} {} со стороной {} и площадью {}".format(
            self._color.colorprop,
            Square.gettype(),
            self._side,
            self.square()
        )
```

-----main.py-----

```
from pdf_gen.pdf_gen import pdf
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square
```

```
def main():
    r = Rectangle(16, 16, "Синий")
    c = Circle(16, "Зеленый")
    s = Square(16, "Красный")
    print(r)
    print(c)
    print(s)
    file = pdf('C:/Users/user/PycharmProjects/RIP/LR2', 'pdfka')
    file.insert_image('C:/Users/user/PycharmProjects/RIP/LR2/102.jpg', width=400,
height=300)
    file.save()

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

Результаты выполнения:

Вывод в консоли:

```
Синий прямоугольник шириной 16, высотой 16 и площадью 256
Зеленый круг радиусом 16 и площадью 804.247719318987
Красный квадрат со стороной 16 и площадью 256
```

```
Process finished with exit code 0
```

Файл, сгенерированный с помощью метода внешнего пакета:

