**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Лабораторная работа №2

“Объектно-ориентированные возможности языка Python.”

Вариант №19

Выполнил:

Студент ИУ5-34Б

Флоринский В. А.

Подпись и дата:

Проверил:

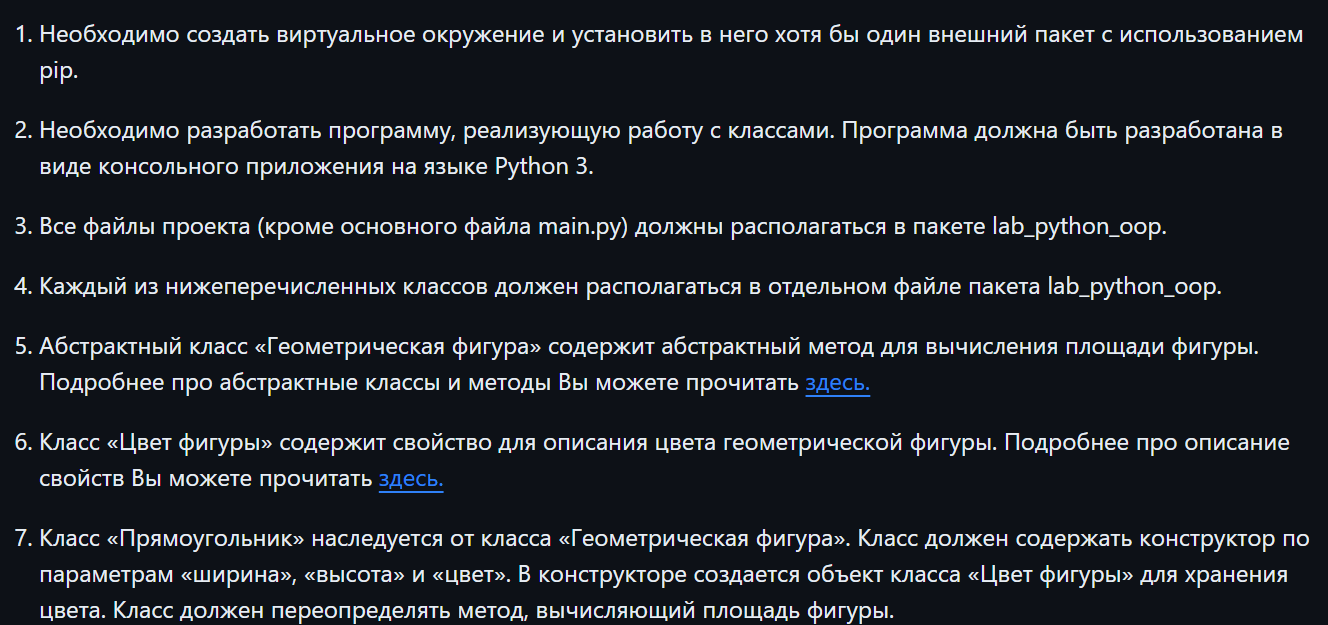
Преподаватель каф. ИУ5

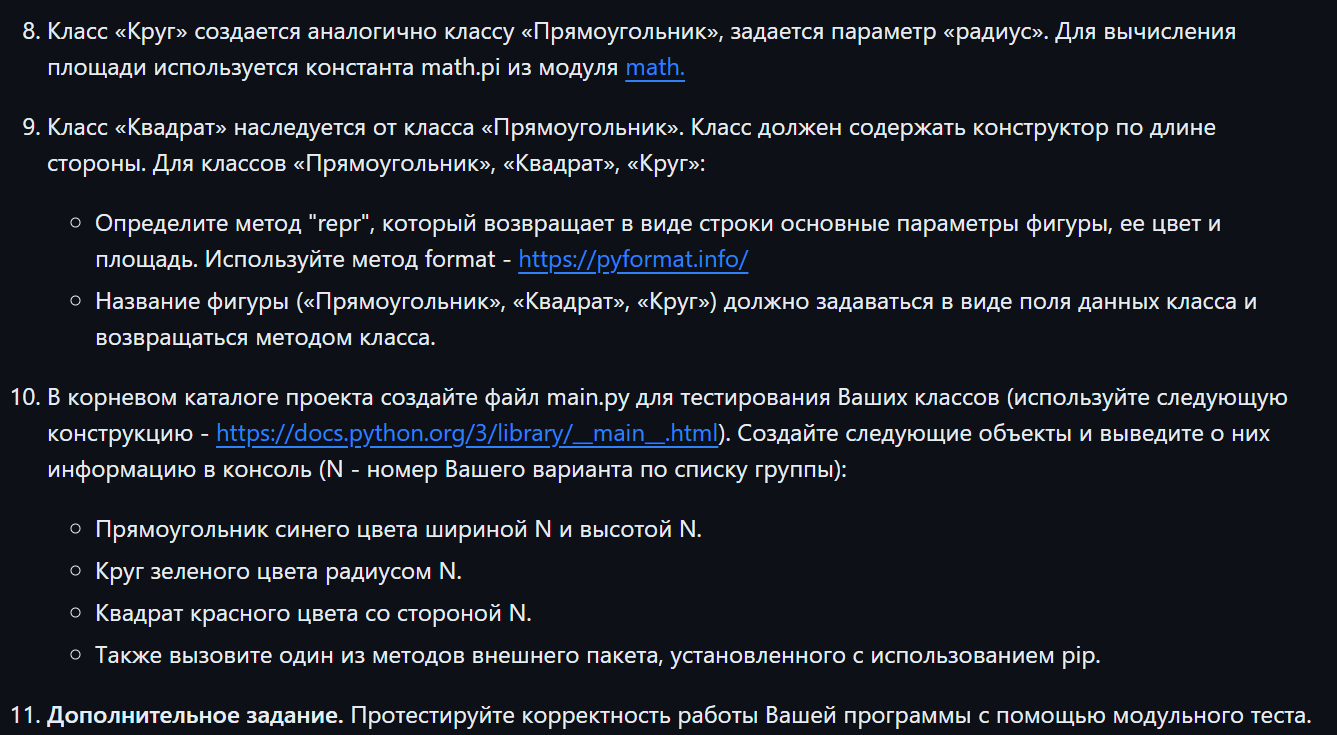
Гапанюк Ю. Е.

Подпись и дата:

Москва, 2023 г.

**Задание.**

****

****

**Текст программ.**

Данная лабораторная работа была создана в виртуальном окружении Python. Программы с классами фигур и цветами должны быть помещены в папку под названием «lab\_python\_oop», а основная программа помещены в папку выше уровнем под названием main.py.

**figure.py:**

from abc import ABC, abstractmethod

class figuresh(ABC):

    @abstractmethod

    def \_\_repr\_\_(self):

        pass

    @abstractmethod

    def square(self):

        pass

**colorcl.py:**

class colorofshape:

    def \_\_init\_\_(self, color):

        self.\_\_color = color

    @property

    def color(self):

        return self.\_\_color

    @color.setter

    def color(self, color):

        self.\_\_color = color

**circlecl.py:**

from lab\_python\_oop.figure import figuresh

from lab\_python\_oop.colorcl import colorofshape

from math import pi

class circle(figuresh):

    name = "circle"

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

        self.\_\_radius = radius

        self.\_\_color = colorofshape(color).color

    def \_\_repr\_\_(self):

        return (f"name: {self.name} | radius : {self.\_\_radius}  | color : {self.\_\_color}\

                 | square : {int(pi\*\*2 \* self.\_\_radius)}")

    def square(self):

        return pi\*\*2 \* self.\_\_radius

**rectanglecl.py:**

from lab\_python\_oop.figure import figuresh

from lab\_python\_oop.colorcl import colorofshape

class rectangle(figuresh):

    name = "rectangle"

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

        self.\_\_width = width

        self.\_\_height = height

        self.\_\_color = colorofshape(color).color

    def square(self):

        return self.\_\_width \* self.\_\_height

    def \_\_repr\_\_(self):

        return (f"name: {self.name} | width : {self.\_\_width} | height: {self.\_\_height} \

                | color : {self.\_\_color} | square : {self.\_\_width \* self.\_\_height}")

**squaresp.py:**

from lab\_python\_oop.rectanglecl import rectangle

from lab\_python\_oop.colorcl import colorofshape

class squareshape(rectangle):

    name = "square"

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

        self.\_\_width = width

        self.\_\_color = colorofshape(color).color

    def \_\_repr\_\_(self):

        return (f"name: {self.name} | width : {self.\_\_width}  | color : {self.\_\_color} \

                | square : {self.\_\_width\*\*2}")

    def square(self):

        return self.\_\_width\*\*2

**main.py:**

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

    from lab\_python\_oop.rectanglecl import rectangle

    from lab\_python\_oop.circlecl import circle

    from lab\_python\_oop.squaresp import squareshape

    from termcolor  import colored

    rec = rectangle(19, 19, "blue")

    cir = circle(19, "green")

    sq = squareshape(19, "red")

    print(colored(rec, "blue"))

    print(colored(cir, "green"))

    print(colored(sq, "red"))

**Примеры работы.**

