

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

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «БКИТ»

Отчет по лабораторной работе №2
«Объектно-ориентированные возможности языка Python»

Выполнил:

студент группы ИУ5-35Б
Большаков Георгий

Подпись и дата:

Проверил:

преподаватель каф. ИУ5
Нардид А.Н.

Подпись и дата:

Москва, 2022 г.

Описание задания

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

Необходимо разработать программу, реализующую работу с классами. Программа должна быть разработана в виде консольного приложения на языке Python 3.

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

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

Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры. Подробнее про абстрактные классы и методы Вы можете прочитать [здесь](#).

Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать [здесь](#).

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.

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

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.

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.

Дополнительное задание. Протестируйте корректность работы Вашей программы с помощью модульного теста.

Текст программы

main.py

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.square import Square
from lab_python_oop.circle import Circle
import cowsay

def main():
    r = Rectangle(3, 6, "Синий")
    c = Circle(5, "Зеленый")
    s = Square(3, "Красный")

    cowsay.cat(str(r) + '\n' + str(c) + '\n' + str(s))

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

Файлы пакета lab_python_oop:

circle.py

```
from lab_python_oop.figure import Figure
from lab_python_oop.color import FigureColor
from math import pi

class Circle(Figure):
    fType = "круг"
    @classmethod
    def getType(cls):
        return cls.fType

    def __init__(self, rad, col):
        self.r = rad
        self.fc = FigureColor()
        self.fc.colorproperty = col

    def square(self):
        return pi * pow(self.r, 2)
```

```

def __repr__(self):
    return '{} {} радиусом {} и площадью {:.3f}'.format(
        self.fc.colorproperty,
        self.getType(),
        self.r,
        self.square()
    )

```

color.py

```

class FigureColor:
    #цвет фигуры

    def __init__(self):
        self._color = None

    @property
    #getter
    def colorproperty(self):
        return self._color

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

```

figure.py

```

from abc import ABC, abstractmethod

class Figure(ABC):
    #Абстрактный класс «Геометрическая фигура»

    @abstractmethod
    def square(self):
        #содержит виртуальный метод для вычисления площади фигуры.

    pass

```

rectangle.py

```

from lab_python_oop.figure import Figure
from lab_python_oop.color import FigureColor

class Rectangle(Figure):

    fType = "прямоугольник"
    @classmethod
    def getType(cls):
        return cls.fType

    def __init__(self, length, width, col):
        self.l = length
        self.w = width
        self.fc = FigureColor()
        self.fc.colorproperty = col

    def square(self):
        return self.l * self.w

```

```

def __repr__(self):
    return '{} {} со сторонами {} и {}, площадью {}'.format(
        self.fc.colorproperty,
        self.getType(),
        self.w,
        self.l,
        self.square()
    )

```

square.py

```

from lab_python_oop.figure import Figure
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.color import FigureColor

class Square(Rectangle):

    fType = "квадрат"
    @classmethod
    def getType(cls):
        return cls.fType

    def __init__(self, sd, col):
        self.side = sd
        super().__init__(self.side, self.side, col)

    def square(self):
        return pow(self.side, 2)

    def __repr__(self):
        return '{} {} со стороной {} и площадью {}'.format(
            self.fc.colorproperty,
            self.getType(),
            self.side,
            self.square()
        )

```

Экранные формы

