



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

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»
Отчёт по лабораторной работе №2
“Объектно-ориентированные возможности языка Python”

Выполнил:

студент группы ИУ5-32Б

Милевич Артём

Подпись и дата:

Проверил:

преподаватель каф. ИУ5

Гапанюк Ю.Е.

Подпись и дата:

Москва, 2021 г.

Описание задания

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` - <https://pyformat.info/>

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 (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 11 и высотой 11.

Круг зеленого цвета радиусом 11.

Квадрат красного цвета со стороной 11.

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

Текст программы

Файл **main.py**:

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square
import pandas as pd

n = 11
```

```

def main():
    r = Rectangle("синий", n, n)
    c = Circle("зеленый", n)
    s = Square("красный", n)

    table = pd.DataFrame(
        {
            "Тип фигуры": [
                Rectangle.get_figure_type(),
                Circle.get_figure_type(),
                Square.get_figure_type(),
            ],
            "Цвет": [
                r.fc.colorproperty, c.fc.colorproperty, s.fc.colorproperty,
            ],
            "Ширина": [
                r.width, '0' , s.width,
            ],
            "Длина": [
                r.height, '0' , s.height,
            ],
            "Радиус": [
                '0', c.r, '0',
            ],
            "Площадь": [
                round(r.square(), 2), round(c.square(), 2) , round(s.square(), 2),
            ]
        }
    )

    print(table)

if __name__ == "__main__":
    # execute only if run as a script
    main()

```

Папка **lab_python_oop**:

Файл **__init__.py**:

```
print('__init__.py')
```

Файл **circle.py**:

```

from .figure import Figure
from .color import FigureColor

```

```

import math

class Circle(Figure):
    """
    Класс «Круг» наследуется от класса «Геометрическая фигура».
    """
    FIGURE_TYPE = "Круг"

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

    def __init__(self, color, radius):
        """
        Класс должен содержать конструктор по параметрам «радиус» и «цвет». В конст
рукторе создается объект класса «Цвет фигуры» для хранения цвета.
        """
        self.r = radius
        self.fc = FigureColor()
        self.fc.colorproperty = color

    def square(self):
        """
        Класс должен переопределять метод, вычисляющий площадь фигуры.
        """
        return math.pi*(self.r**2)

    def __repr__(self):
        return '{} {} цвета радиусом {} площадью {}'.format(
            Circle.get_figure_type(),
            self.fc.colorproperty,
            self.r,
            self.square()
        )

```

Файл **color.py**:

```

class FigureColor:
    """
    Класс «Цвет фигуры»
    """

    def __init__(self):
        self._color = None

    @property
    def colorproperty(self):
        """

```

```

        Get-аксессор
        """
        return self._color

    @colorproperty.setter
    def colorproperty(self, value):
        """
        Set-аксессор
        """
        self._color = value

```

Файл **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

```

Файл **rectangle.py**:

```

from .figure import Figure
from .color import FigureColor

class Rectangle(Figure):
    """
    Класс «Прямоугольник» наследуется от класса «Геометрическая фигура».
    """
    FIGURE_TYPE = "Прямоугольник"

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

    def __init__(self, color, width, height):
        """
        Класс должен содержать конструктор по параметрам «ширина», «высота» и «цвет
        ». В конструкторе создается объект класса «Цвет фигуры» для хранения цвета.
        """
        self.width = width

```

```

        self.height = height
        self.fc = FigureColor()
        self.fc.colorproperty = color

    def square(self):
        """
        Класс должен переопределять метод, вычисляющий площадь фигуры.
        """
        return self.width*self.height

    def __repr__(self):
        return '{} {} цвета шириной {} и высотой {} площадью {}.'.format(
            Rectangle.get_figure_type(),
            self.fc.colorproperty,
            self.width,
            self.height,
            self.square()
        )

```

Файл **sqare.py**:

```

from .rectangle import Rectangle

class Square(Rectangle):
    """
    Класс «Квадрат» наследуется от класса «Прямоугольник».
    """
    FIGURE_TYPE = "Квадрат"

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

    def __init__(self, color, side):
        """
        Класс должен содержать конструктор по параметрам «сторона» и «цвет».
        """
        self.side = side
        super().__init__(color, self.side, self.side)

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

```

Результат выполнения программы

```
__init__.py
    Тип фигуры      Цвет  Ширина  Длина  Радиус  Площадь
0  Прямоугольник    синий   11    11      0    121.00
1      Круг         зеленый   0     0     11    380.13
2      Квадрат      красный  11    11      0    121.00
(.virtual) PS C:\Microsoft VS Code\python\lab_2\code>
```