

**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 ИУ5-32Б

Лялько Никита

Подпись и дата:

Проверил:

преподаватель каф. ИУ5

Гапанюк Ю.Е.

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 `getr`, 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

```
1 from lab_python_oop.rectangle import Rectangle
2 from lab_python_oop.circle import Circle
3 from lab_python_oop.square import Square
4 from prettytable import PrettyTable
5
6 N = 15
7
8 def main():
9     r = Rectangle(N, N, "синего")
10    c = Circle(N, "зеленого")
11    s = Square(N, "красного")
12    print(r)
13    print(c)
14    print(s)
15
16    x = PrettyTable()
17    x.field_names = ["Название фигуры", "Цвет фигуры", "Площадь фигуры"]
18    x.add_rows([
19        ["Прямоугольник", r._figure_color.colorproperty, r.get_area()],
20        ["Круг", c._figure_color.colorproperty, c.get_area()],
21        ["Квадрат", s._figure_color.colorproperty, s.get_area()]
22    ])
23    print(x)
24
25 if __name__ == "__main__":
26     main()
```

Файлы пакета lab_python_oop:
circle.py

```

1  from lab_python_oop.figure import Figure
2  from lab_python_oop.figure_color import Figure_color
3  import math
4
5
6  class Circle(Figure):
7      """
8          Класс «Круг» наследуется от класса «Геометрическая фигура».
9          """
10     FIGURE_TYPE = 'Круг'
11
12     @classmethod
13     def get_figure_type(cls):
14         return cls.FIGURE_TYPE
15
16     def __init__(self, radius, color):
17
18         self._radius = radius
19         self._figure_color = Figure_color()
20         self._figure_color.colorproperty = color
21
22     def get_area(self):
23
24         return math.pi * (self._radius ** 2)
25
26     def __repr__(self):
27         return '{} {} цвета радиусом {} площадью {}'.format(
28             Circle.get_figure_type(),
29             self._figure_color.colorproperty,
30             self._radius,
31             self.get_area()
32

```

figure_color.py

```

1 class Figure_color:
2     """
3     Класс «Цвет фигуры»
4     """
5
6     def __init__(self):
7         self._color = None
8
9     @property
10    def colorproperty(self):
11        """
12        Get-аксессор
13        """
14        return self._color
15
16    @colorproperty.setter
17    def colorproperty(self, value):
18        """
19        Set-аксессор
20        """
21        self._color = value

```

figure.py

```

1 from abc import ABC, abstractmethod
2 class Figure(ABC):
3     """
4     Абстрактный класс «Геометрическая фигура»
5     """
6
7     @abstractmethod
8     def get_area(self):
9         """
10        содержит виртуальный метод для вычисления площади фигуры.
11        """
12        pass

```

rectangle.py

```

1  from lab_python_oop.figure import Figure
2  from lab_python_oop.figure_color import Figure_color
3  class Rectangle(Figure):
4      """
5          Класс «Прямоугольник» наследуется от класса «Геометрическая фигура»
6      """
7      FIGURE_TYPE = "Прямоугольник"
8
9      @classmethod
10     def get_figure_type(cls):
11         return cls.FIGURE_TYPE
12
13     def __init__(self, width, height, color):
14
15         self._figure_color = Figure_color()
16         self._width = width
17         self._height = height
18         self._figure_color.colorproperty = color
19
20     def get_area(self):
21
22         return self._width * self._height
23
24     def __repr__(self):
25         return '{} {} цвета шириной {} и высотой {} площадью {}.'.format(
26             Rectangle.get_figure_type(),
27             self._figure_color.colorproperty,
28             self._width,
29             self._height,
30             self.get_area()
31         )

```

square.py

```

1      from lab_python_oop.rectangle import Rectangle
2
3      class Square(Rectangle):
4          """
5              Класс «Квадрат» наследуется от класса «Прямоугольник».
6          """
7          FIGURE_TYPE = "Квадрат"
8
9          @classmethod
10         def get_figure_type(cls):
11             return cls.FIGURE_TYPE
12
13         def __init__(self, side, color):
14             """
15                 Класс должен содержать конструктор по параметрам «сторона» и «цвет»
16             """
17             self._side = side
18             super().__init__(side, side, color)
19
20         def __repr__(self):
21             return '{} {} цвета со стороной {} площадью {}.'.format(
22                 Square.get_figure_type(),
23                 self._figure_color.colorproperty,
24                 self._side,
25                 self.get_area()
26             )

```

экранные формы

```

C:\Users\lyalk\PycharmProjects\pythonProject3\venv\Scripts\python.exe C:\Users\lyalk\PycharmProjects\pythonProject3\main.py
Прямоугольник синего цвета шириной 15 и высотой 15 площадью 225.
Круг зеленого цвета радиусом 15 площадью 706.8583470577034.
Квадрат красного цвета со стороной 15 площадью 225.
+-----+
| Название фигуры | Цвет фигуры | Площадь фигуры |
+-----+
| Прямоугольник | синего | 225 |
| Круг | зеленого | 706.8583470577034 |
| Квадрат | красного | 225 |
+-----+

Process finished with exit code 0

```