



Министерство науки и высшего образования Российской
Федерации Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский государственный технический
университет имени Н.Э. Баумана
(национальный исследовательский
университет)» (МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ Информатика и системы управления

КАФЕДРА _____ Системы обработки информации и управления

Лабораторная работа №2
По курсу «Разработка интернет приложений»

Подготовила:

Студентка группы ИУ5-55Б.

Очеретная С.В.

30.09.2020

Проверил:

Преподаватель кафедры ИУ5

Гапанюк Ю.Е.

Москва, 2021 г.

Описание задания:

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.
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.
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.
 - 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`.

Текст программы:

Абстрактный класс «Геометрическая фигура», содержащий абстрактный метод для вычисления площади фигуры:

File "figure.py":

```
from abc import ABC, abstractmethod
```

```
class Figure(ABC):
```

```
    @abstractmethod
```

```
    def square(self):
```

```
        pass
```

Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры:

```
class FigureColor:
```

```
    def __init__(self):
```

```
        self._color = None
```

```
    @property
```

```
    def color(self):
```

```
        return self._color
```

```
    @color.setter
```

```
    def color(self, value):
```

```
        self._color = value
```

```
    @color.deleter
```

```
    def color(self):
```

```
        del self._color
```

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:

```
class Rectangle(GeomFigure):
```

```
    NAME = "Прямоугольник"
```

```
    def __init__(self, _width, _height, _color):
```

```
        self.width = _width
```

```
        self.height = _height
```

```
        self.color = FigureColor()
```

```

        self.color.color = _color

def square(self):
    return self.width * self.height

@classmethod
def get_name(obj):
    return obj.NAME

def __repr__(self):
    return 'Фигура: {} \nПараметры: \n\tширина: {} \n\tвысота: {} \n\tцвет: {} \n\tплощадь: {} \n'.format(
        Rectangle.get_name(),
        self.width,
        self.height,
        self.color.color,
        self.square()
    )

```

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

```

class Circle(GeomFigure):
    NAME = "Круг"

    def __init__(self, _radius, _color):
        self.radius = _radius
        self.color = FigureColor()
        self.color.color = _color

    def square(self):
        return self.radius * self.radius * math.pi

@classmethod
def get_name(obj):
    return obj.NAME

def __repr__(self):
    return 'Фигура: {} \nПараметры: \n\tрадиус: {} \n\t

```

```

цвет: {} \n \t площадь: {} \n'.format(
    Circle.get_name(),
    self.radius,
    self.color.color,
    self.square()
)

```

Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны

```
class Square(GeomFigure):
```

```
    NAME = "Квадрат"
```

```
    def __init__(self, _side, _color):
```

```
        self.side = _side
```

```
        self.color = FigureColor()
```

```
        self.color.color = _color
```

```
    def square(self):
```

```
        return self.side * self.side
```

```
    @classmethod
```

```
    def get_name(obj):
```

```
        return obj.NAME
```

```
    def __repr__(self):
```

```
        return 'Фигура: {} \n \t Параметры: \n \t ширина стороны: {} \n \
```

```
цвет: {} \n \t площадь: {} \n'.format(
```

```
    Square.get_name(),
```

```
    self.side,
```

```
    self.color.color,
```

```
    self.square()

```

```
)
```

Для классов «Прямоугольник», «Квадрат», «Круг»: определен метод "repr", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь.

Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращается методом класса

Main.py для тестирования:

```
from lab_python_oop.square import Square
from lab_python_oop.circle import Circle
from lab_python_oop.rectangle import Rectangle
from colorama import init, Fore
init()
```

```
def main():
    rect = Rectangle(8, 8, "синий")
    circle = Circle(8, "зеленый")
    square = Square(8, "красный")
    print(Fore.BLUE)
    print(rect)
    print(Fore.GREEN)
    print(circle)
    print(Fore.RED)
    print(square)
```

Экранные формы с примерами работы программы

```
(webappdevelopment-QG1KFBN) C:\Users\Pocht\OneDrive\Study\5 семестр\РИП\webappdevelopment\lab2>python main.py
Фигура: Прямоугольник
Параметры:
    ширина: 8
    высота: 8
    цвет: синий
    площадь: 64

Фигура: Круг
Параметры:
    радиус: 8
    цвет: зеленый
    площадь: 201.06192982974676

Фигура: Квадрат
Параметры:
    ширина стороны: 8
    цвет: красный
    площадь: 64
```