



ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Системы обработки информации и управления» (ИУ-5)

ЛАБОРАТОРНАЯ РАБОТА

№ 2

Объектно-ориентированные возможности языка Python

Группа ИУ5-35Б

Студент  16.12.2024 /Д.Е. Мушкарин /

(Подпись, дата)

(И.О.Фамилия)

Преподаватель _____ /Ю. Е. Гапанюк/

(Подпись, дата)

(И.О.Фамилия)

2024

Задание:

- 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 `"get"`, 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`.

Текст программы (Python):

main.py

```
from Square import Square
from Circle import Circle
from Rectangle import Rectangle
import pyfiglet # type: ignore

print(pyfiglet.figlet_format("laba!", font="slant"))
```

```
def main():
    Rectangle(13,13, "синий").repr()
    Circle(13, "зеленый").repr()
    Square(13, "красный").repr()
```

```
if __name__ == "__main__":
```

```
    main()
```

Color.py

```
class Color:
    def __init__(self, color : str):
        self.color = color
```

```
    @property
```

```
    def color(self) -> str:
        return self._color
```

```
    @color.setter
```

```
    def color(self, color : str):
        self._color = color
```

AbstractShape.py

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

```
class AbstractShape(ABC):
    @abstractmethod
    def area(self):
        pass
```

Square.py

```
from Rectangle import Rectangle
```

```

class Square(Rectangle):
    name = "Квадрат"
    def __init__(self, side : float, color : str):
        super().__init__(side, side, color)
    def area(self) -> float:
        return self._width * self._height
    def get_name(cls) -> str:
        return cls.name
    def repr(self):
        print(f"{self.get_name()} со стороной {self._width}, площадью {self.area()}, {self._color.color} цвет")

```

Rectangle.py

```

from Color import Color
from AbstractShape import AbstractShape

```

```

class Rectangle(AbstractShape):
    name = "Прямоугольник"
    def __init__(self, width : float, height : float, color : str):
        self._width = width
        self._height = height
        self._color = Color(color)

    def area(self) -> float:
        return self._width * self._height
    def get_name(cls) -> str:
        return cls.name

    def repr(self):
        print(f"{self.get_name()} ширины {self._width}, высоты {self._height}, площадью {self.area()}, {self._color.color} цвет")

```

Circle.py

```

from Color import Color
from AbstractShape import AbstractShape
from math import pi

```

```

class Circle(AbstractShape):
    name = "Круг"
    def __init__(self, radius : float, color : str):
        self._radius = radius
        self._color = Color(color)

    def area(self) -> float:
        return pi*self._radius**2
    def get_name(cls) -> str:
        return cls.name

    def repr(self):
        print(f"{self.get_name()} с радиусом {self._radius}, площадью {self.area():.4f}, {self._color.color} цвет")

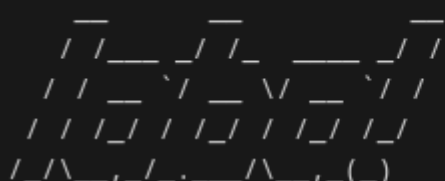
```

Вывод:

```

● └─[$] <git:(main*)> /home/user/Documents/py_sem3/lab_python_oop

```



```

Прямоугольник ширины 13, высоты 13, площадью 169, синий цвет
Круг с радиусом 13, площадью 530.9292, зеленый цвет
Квадрат со стороной 13, площадью 169, красный цвет

```