

**Отчёт по лабораторной работе №2**  
**«Python. Объектно-ориентированные возможности»**  
**По курсу «Разработка интернет приложений»**

## 1.Задание.

Необходимо разработать программу, реализующую работу с классами.

1.1. Программа должна быть разработана в виде консольного приложения на языке Python 3.

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

lab\_python\_oop.

1.3. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета lab\_python\_oop.

1.3.1. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры.

1.3.2. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.

1.3.3. 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.

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

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

1.3.6. Для классов «Прямоугольник», «Квадрат», «Круг»:

1.3.6.1. Определите метод "getr", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь.

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

1.4. В корневом каталоге проекта создайте файл main.py для тестирования Ваших классов.

Создайте следующие объекты и выведите о них информацию в консоль:

1.4.1. Прямоугольник синего цвета шириной 3 и высотой 2.

1.4.2. Круг зеленого цвета радиусом 5.

1.4.3. Квадрат красного цвета со стороной 5.

## 2. Исходный код.

2.1. Класс «Геометрическая фигура»:

```
from abc import ABC, abstractmethod
class GeomFigure(ABC):
    @abstractmethod
    def AreaCalc(self):
        return self.area
```

2.2. Класс «Цвет фигуры»:

```
class Color:
    def __init__(self):
        self._color = None

    def getcolor(self):
        return self._color

    def setcolor(self, value):
        self._color = value

    def delcolor(self):
        del self._color

    color = property(getcolor, setcolor, delcolor, "I'm the 'color' property.")
```

2.3. Класс «Прямоугольник»:

```
class Rectangle(GeomFigure):
    def __init__(self,w,h):
        self.FigureType = "Rectangle"
        print("Enter figure color")
        self.width=w
        self.height=h
        self.color = Color()
        self.color.setcolor(input())

    def AreaCalc(self):
        return float(self.width)*float(self.height)
```

```

    def __repr__(self):
        return "Figure data:   Figure type:{}; Width:{}; Height:{}; Color:{}; Area:{};"
        .format( self.FigureType, self.width, self.height,
self.color.getcolor(), self.AreaCalc())

```

#### 2.4. Класс «Квадрат»:

```

class Square(Rectangle):
    def __init__(self,s):
        super().__init__(s,s)
        self.side=s
        self.FigureType="Square"

    def __repr__(self):
        return "Figure data:   Figure type:{}; Side:{}; Color:{}; Area:{};"
        .format( self.FigureType, self.side, self.color.getcolor(),
self.AreaCalc())

```

#### 2.5. Класс «Круг»:

```

class Circle(GeomFigure):
    def __init__(self,r):
        self.FigureType = "Circle"
        self.radius=r
        print("Enter figure color")
        self.color = Color()
        self.color.setcolor(input())

    def AreaCalc(self):
        import math
        return float(self.radius)*float(self.radius)*math.pi

    def __repr__(self):
        return "Figure data:   Figure type:{}; Radius:{}; Color:{}; Area:{};"
        .format(
self.FigureType,self.radius,self.color.getcolor(),self.AreaCalc())

```

#### 2.5. \_\_main\_\_:

```

def NumInput(v):
    i=False
    while i==False:
        print("Enter", v)
        a = input()
        try:
            float(a)
            i=True
        except ValueError:
            i=False
            print(a, "is not a number.")
    return a

j=False
while j==False:
    print("Chose figure")
    print("Rectangle - 1")
    print("Square - 2")
    print("Circle - 3")
    rezh=input()
    if rezh=="1" or rezh=="2" or rezh=="3":
        j=True
    else:
        print("Wrong number of figure")
        j=False
if rezh=="1":
    width = NumInput("width")
    height = NumInput("height")
    R1 = Rectangle(width,height)
    print(R1)
if rezh=="2":
    side = NumInput("side")
    S1 = Square(side)

```

```

        print(S1)
if rezh=="3":
    radius=NumInput("radius")
    C1 = Circle(radius)
    print(C1)    print(S1.__repr__())
if rezh=="3":
    C1 = Circle()
    print(C1.__repr__())

```

### 3. Скриншоты с результатами выполнения.

#### 3.1. Прямоугольник синего цвета шириной 3 и высотой 2.

```

Chose figure
Rectangle - 1
Square - 2
Circle - 3
1
This figure is Rectangle
Enter width
3
Enter height
2
Enter figure color
Синий
Figure data:  Figure type:Rectangle; Width:3; Height:2; Color:Синий; Area:6;

Process finished with exit code 0

```

#### 3.2. Круг зеленого цвета радиусом 5.

```

Chose figure
Rectangle - 1
Square - 2
Circle - 3
3
This figure is Circle
Enter radius
5
Enter figure color
Зелёный
Figure data:  Figure type:Circle; Radius:5; Color:Зелёный; Area:78.53981633974483;

Process finished with exit code 0

```

#### 3.3. Квадрат красного цвета со стороной 5.

```

Chose figure
Rectangle - 1
Square - 2
Circle - 3
2
This figure is Square
Enter side
5
Enter figure color
Красный
Figure data:  Figure type:Square; Side:5; Color:Красный; Area:25;

Process finished with exit code 0

```