

```
from lab3.Rectangle import Rectangle
from lab3.Square import Square
from lab3.Circle import Circle
from lab3.Color import Color
import requests
import json

def get_time():
    response = requests.get("http://worldtimeapi.org/api/timezone/Europe/Moscow")
    return json.loads(response.text)["datetime"]

print("Current time: " + get_time()+"\n")

color = Color(255, 0, 255)
rect = Rectangle(10, 13, color)
print(rect)

square = Square(11, color)
print(square)

circle = Circle(10, color)
print(circle)
from lab3.GeometricFigure import Figure
from lab3.Color import Color
import math

class Circle(Figure):

    name = 'Circle'

    def __init__(self, radius, color):
        self.color = color
        self.radius = radius
```

```

    def calc_square(self):
        return math.pi*(self.radius**2)

    @classmethod
    def get_name(self):
        return self.name

    def __repr__(self):
        return ("Name:  {};  radius:  {};  RGB:  {}, {}, {};  square:
 {}".format(self.get_name(),  str(self.radius),  str(self.color.red),  str(self.color.green),
 str(self.color.blue), str(self.calc_square()))))
class Color:
    def __init__(self, red, green, blue):
        self.red = red
        self.green = green
        self.blue = blue
    from abc import ABC, abstractmethod

class Figure(ABC):
    @abstractmethod
    def calc_square(self):
        pass
    from lab3.Color import Color
    from lab3.GeometricFigure import Figure

class Rectangle(Figure):

    name = 'Rectangle'

    def __init__(self, width, height, color):
        self.width = width
        self.height = height
        self.color = color

```

```

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

    @classmethod
    def get_name(self):
        return self.name

    def __repr__(self):
        return ("Name: {}; width: {}; height: {}; RGB: {},{},{}; square:
{}".format(self.get_name(),    str(self.width),    str(self.height),    str(self.color.red),
str(self.color.green), str(self.color.blue), str(self.calc_square()))
from lab3.Rectangle import Figure

class Square(Figure):

    name = 'Square'

    def __init__(self, length, color):
        self.color = color
        self.length = length

    def calc_square(self):
        return self.length**2

    @classmethod
    def get_name(self):
        return self.name

    def __repr__(self):
        return ("Name: {}; length: {}; RGB: {},{},{}; square:
{}".format(self.get_name(),    str(self.length),    str(self.color.red),    str(self.color.green),
str(self.color.blue), str(self.calc_square()))

```