

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «ПИКяП»

Отчет по лабораторной работе №2

«Объектно-ориентированные возможности языка Python»

Выполнил:

студент группы ИУ5-23Б

Илюхин И. Д.

Подпись и дата: 11.12.2024

Проверил:

преподаватель каф. ИУ5

Нардид А. И.

Подпись и дата:

Москва, 2024 г.

Цель лабораторной работы: изучение объектно-ориентированных возможностей языка Python.

Задание:

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. Подробнее про абстрактные классы и методы Вы можете прочитать [здесь](#).
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать [здесь](#).
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. Используйте метод `format` - <https://pyformat.info/>
 - 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 (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`.

Файл main.py

```
import sys
sys.path.append('Lab2_python/lab_python_oop')
from lab_python_oop.square import Square
from lab_python_oop.rectangle import Rect
from lab_python_oop.circle import Circle

square = Square(7, "red")
circle = Circle(7, "blue")
rectangle = Rect(7, 5, "green")
print(square)
print(circle)
print(rectangle)
```

Файл circle.py

```
from geometric_shape import Geometric_Shape
import math

class Circle(Geometric_Shape):
    name = "circle"
    def __init__(self, radius, color):
        self.radius = radius
        self.color = color

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

    def __repr__(self):
        return "Shape - {}, radius - {}, color - {}, area - {}".format(self.name,
self.radius, self.color, self.calculate())
```

Файл color.py

```
from geometric_shape import Geometric_Shape

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

    def calculate(self):
        return self.color
```

Файл geometric_shape.py

```
from abc import ABC, abstractmethod

class Geometric_Shape(ABC):
    @abstractmethod
    def calculate(self):
        pass
```

Файл rectangle.py

```
from geometric_shape import Geometric_Shape

class Rect(Geometric_Shape):
    name = "rectangle"
    def __init__(self, height, width, color):
        self.height = height
        self.width = width
        self.color = color

    def calculate(self):
        return self.height * self.width
    def __repr__(self):
        return "Shape - {}, color - {}, height - {}, width - {}, area - {}".format(self.name, self.color, self.height, self.width, self.calculate())
```

Файл square.py

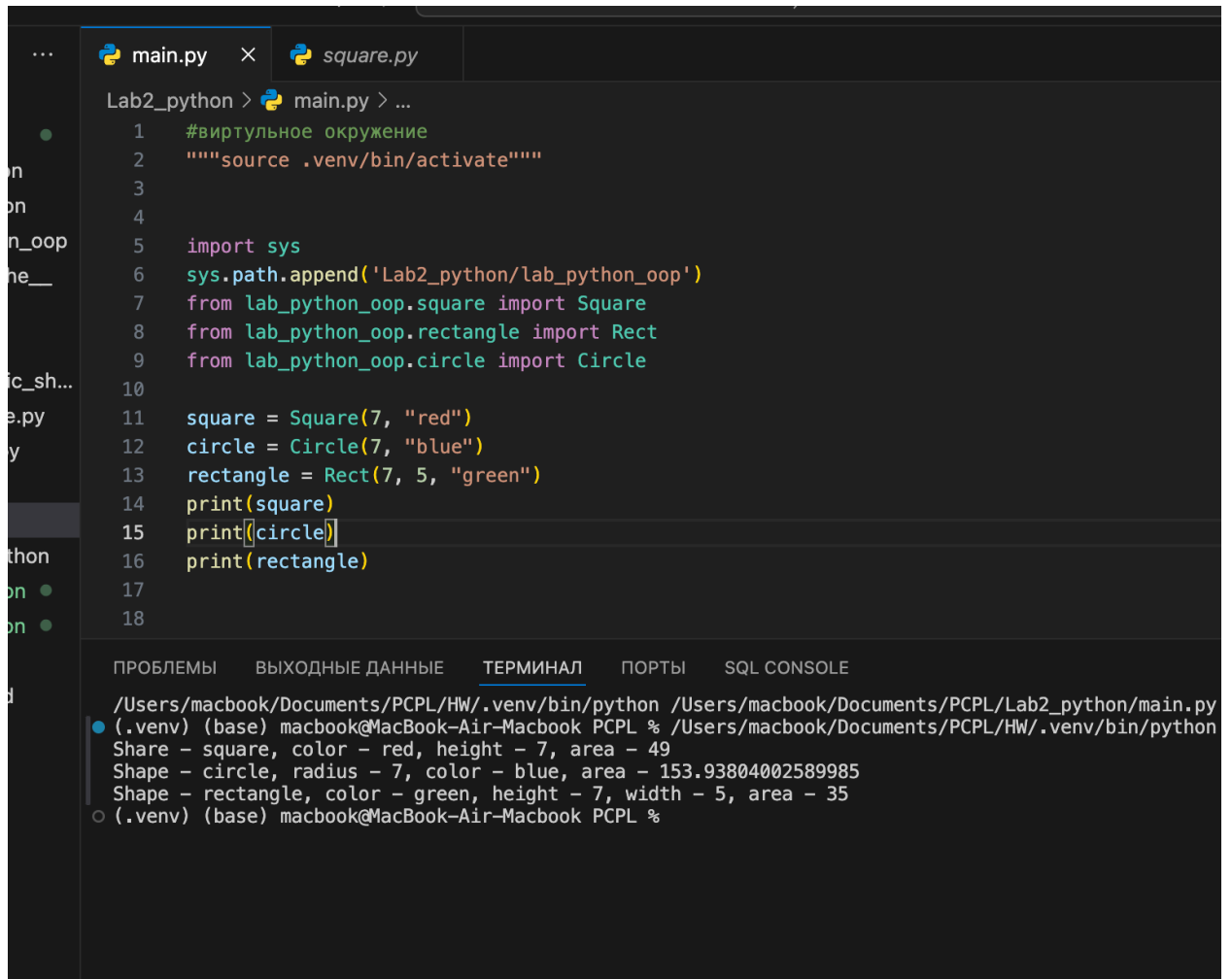
```
from rectangle import Rect
from geometric_shape import Geometric_Shape

class Square(Rect):
    name = "square"
    def __init__(self, height, color):
        self.height = height
        self.color = color

    def calculate(self):
        return self.height**2

    def __repr__(self):
        return "Share - {}, color - {}, height - {}, area - {}".format(self.name, self.color, self.height, self.calculate())
```

Пример работы программы:



The image shows a code editor with two tabs: `main.py` and `square.py`. The `main.py` tab is active, displaying a Python script. The script starts with a comment in Russian and a command to activate a virtual environment. It then imports the `sys` module and appends the path to the `lab_python_oop` directory. Next, it imports the `Square`, `Rect`, and `Circle` classes from their respective modules. The script creates three objects: a red square with side length 7, a blue circle with radius 7, and a green rectangle with height 7 and width 5. Finally, it prints each object, which triggers their `__str__` method.

```
Lab2_python > main.py > ...
1  #виртуальное окружение
2  """source .venv/bin/activate"""
3
4
5  import sys
6  sys.path.append('Lab2_python/lab_python_oop')
7  from lab_python_oop.square import Square
8  from lab_python_oop.rectangle import Rect
9  from lab_python_oop.circle import Circle
10
11  square = Square(7, "red")
12  circle = Circle(7, "blue")
13  rectangle = Rect(7, 5, "green")
14  print(square)
15  print(circle)
16  print(rectangle)
17
18
```

Below the code editor is a terminal window with tabs: `ПРОБЛЕМЫ`, `ВЫХОДНЫЕ ДАННЫЕ`, `ТЕРМИНАЛ` (selected), `ПОРТЫ`, and `SQL CONSOLE`. The terminal shows the command to run the script and its output:

```
/Users/macbook/Documents/PCPL/HW/.venv/bin/python /Users/macbook/Documents/PCPL/Lab2_python/main.py
(.venv) (base) macbook@MacBook-Air-Macbook PCPL % /Users/macbook/Documents/PCPL/HW/.venv/bin/python
Shape - square, color - red, height - 7, area - 49
Shape - circle, radius - 7, color - blue, area - 153.93804002589985
Shape - rectangle, color - green, height - 7, width - 5, area - 35
(.venv) (base) macbook@MacBook-Air-Macbook PCPL %
```