Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



**Отчет**

**По курсу «Разработка интернет приложений»**

**«Лабораторная работа №2»**

**ИСПОЛНИТЕЛЬ:**

Группа ИУ5-53Б

Кокозов С.И.

"18" января 2021 г.

**ПРЕПОДАВАТЕЛЬ:**

Гапанюк Ю.Е

Москва 2020

**Задание:**

### 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.](https://docs.python.org/3/library/abc.html)
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать [здесь.](https://docs.python.org/3/library/functions.html#property)
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.](https://docs.python.org/3/library/math.html)
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.

**Класс Rectangle:**

# Класс прямоугольник

from figure import Figure

from color import FigureColor

class Rectangle(Figure):

FIGURE\_TYPE = "Прямоугольник"

@classmethod

def get\_figure\_type(cls):

return cls.FIGURE\_TYPE

def \_\_init\_\_(self, color\_param, width\_param, height\_param):

self.width = width\_param

self.height = height\_param

self.fc = FigureColor()

self.fc.colorproperty = color\_param

def square(self):

return self.width\*self.height

def \_\_repr\_\_(self):

return '{} цвета: {}, шириной: {}, высотой: {}, и площадью: {}.'.format(

Rectangle.get\_figure\_type(),

self.fc.colorproperty,

self.width,

self.height,

self.square()

)

**Класс Circle:**

# Класс Круг

from figure import Figure

from color import FigureColor

import math

class Circle(Figure):

FIGURE\_TYPE = "Круг"

@classmethod

def get\_figure\_type(cls):

return cls.FIGURE\_TYPE

def \_\_init\_\_(self, color\_param, r\_param):

self.r = r\_param

self.fc = FigureColor()

self.fc.colorproperty = color\_param

def square(self):

return math.pi\*(self.r\*\*2)

def \_\_repr\_\_(self):

return '{} цвета: {}, радиусом: {}, и площадью: {}.'.format(

Circle.get\_figure\_type(),

self.fc.colorproperty,

self.r,

self.square()

)

**Класс Square:**

#Класс квадрат

from rectangle import Rectangle

class Square(Rectangle):

"""

Класс «Квадрат» наследуется от класса «Прямоугольник».

"""

FIGURE\_TYPE = "Квадрат"

@classmethod

def get\_figure\_type(cls):

return cls.FIGURE\_TYPE

def \_\_init\_\_(self, color\_param, side\_param):

"""

Класс должен содержать конструктор по параметрам «сторона» и «цвет».

"""

self.side = side\_param

super().\_\_init\_\_(color\_param, self.side, self.side)

def \_\_repr\_\_(self):

return '{} цвета: {}, стороной: {}, и площадью: {}.'.format(

Square.get\_figure\_type(),

self.fc.colorproperty,

self.side,

self.square()

)

**Класс Figures:**

# Абстрактный класс фигуры

from abc import ABC, abstractmethod

class Figure(ABC):

@abstractmethod

def square(self):

pass

**Файл main.py**

#Кокозов С.И. ИУ-5 53Б ЛАБ2

from rectangle import Rectangle

from circle import Circle

from square import Square

from tkinter import\*

import pickle

def close\_window(root):

root.destroy()

def main():

starts = True

getch = ''

while starts:

r = Rectangle("Желтый", 22, 22)

c = Circle("Зеленый", 7)

s = Square("Синий", 77)

print('MENU:')

putch = input('1 - вывести задание.\n'+

'2 - протестировать подключенную библиотеку(Tkinter).\n'+

'3 - выйти.'+

'\n'+

'\n')

putch = int(putch)

print()

if putch == 3:

starts = False

elif putch == 1:

print(r)

print(c)

print(s)

a = str(r)

print('')

elif putch == 2:

root = Tk()

root.title('Проверка')

root.minsize(800,600)

root.configure(background='#eea132')

L = Label(root,bg='#eea132',font=('Arial Black',22),text='Tkinter. Это Спиридончик))) Спасибо за внимание')

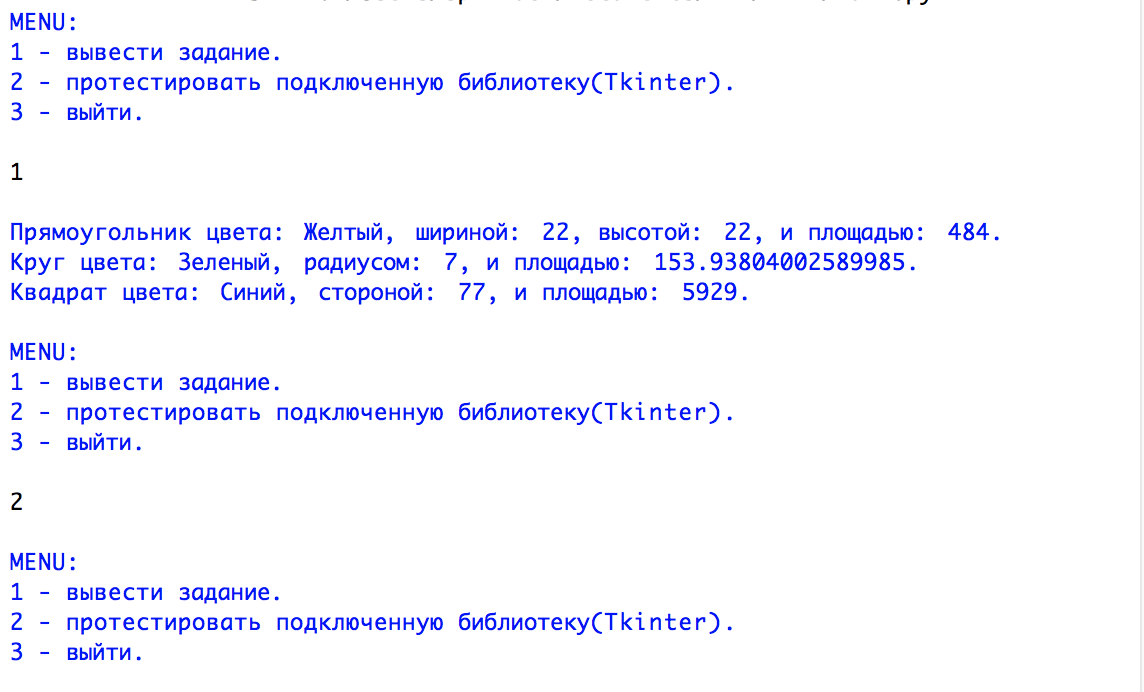
L.place(x=22,y=0)

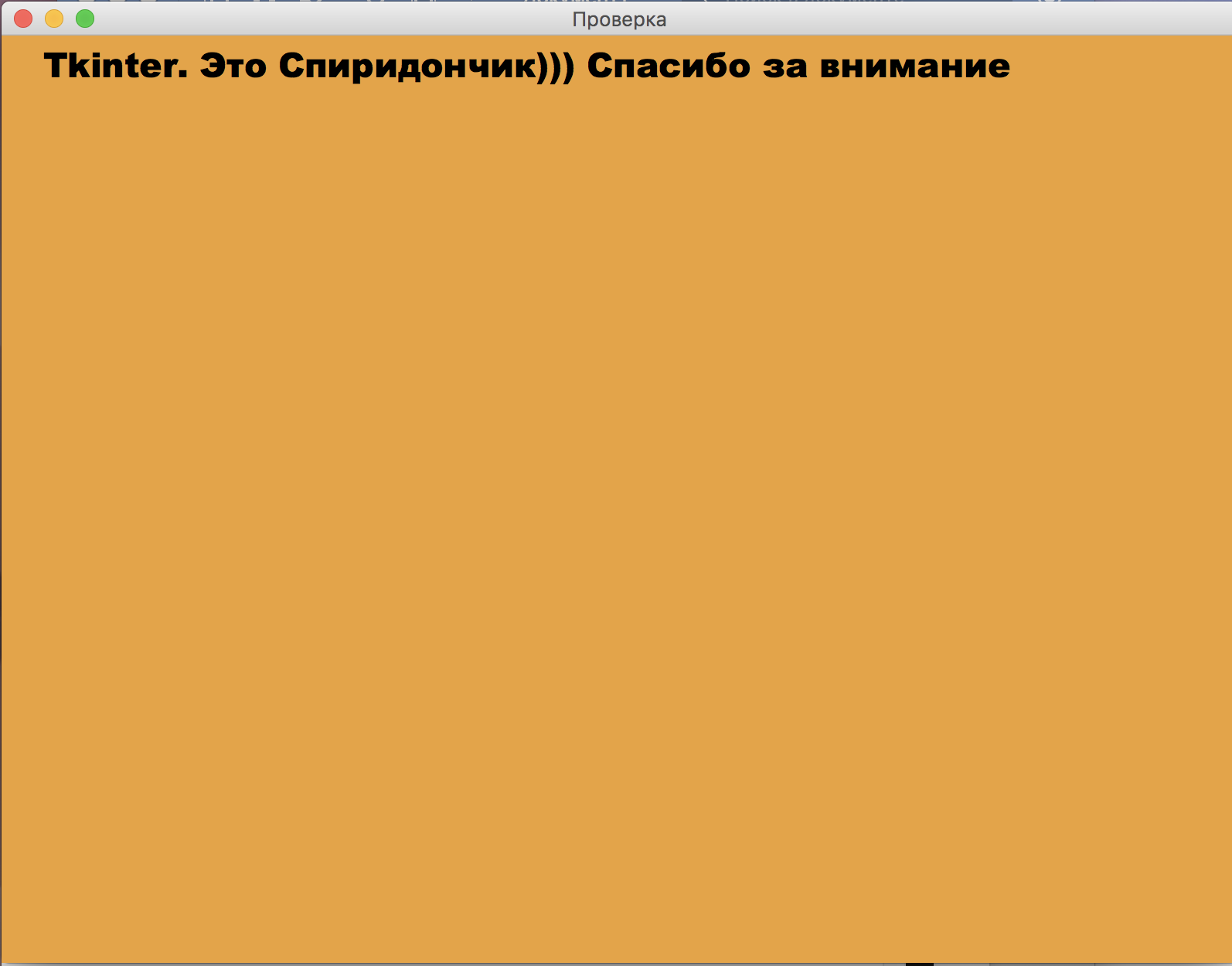
root.mainloop()

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Вывод программы:**

****

****