

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ  
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное  
образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра “Математическая кибернетика и информационные технологии”

**ОТЧЕТ**

по лабораторной работе №5

по дисциплине «Введение в информационные технологии»

Тема: «Работа с классами»

Выполнил: студент группы БВТ2505

Кручко Александр Вадимович

Проверил: Павликов А. Е.

Москва, 2025

## Цель работы

Получить практический опыт работы с ООП в Python.

## Индивидуальное задание

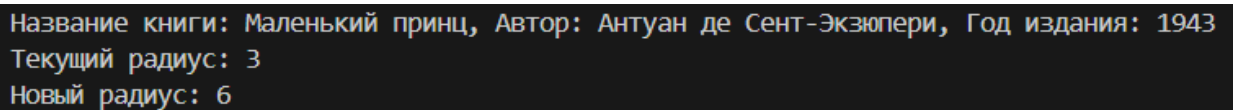
### Задание 1: Базовый класс и методы

1. Определите класс Book, который имеет три атрибута: title (название), author (автор), и year (год издания).
2. Добавьте метод get\_info(), который возвращает информацию о книге в формате: "Название книги: [title], Автор: [author], Год издания: [year]".

### Задание 2: Работа с конструктором

1. Определите класс Circle для представления круга.
2. Используйте конструктор \_\_init\_\_ для инициализации радиуса круга (radius).
3. Добавьте метод get\_radius(), который возвращает значение радиуса.
4. Добавьте метод set\_radius(new\_radius), который позволяет изменить радиус круга.
5. Создайте объект класса Circle, измените его радиус и выведите новый радиус на экран

## Скриншоты выполнения



```
Название книги: Маленький принц, Автор: Антуан де Сент-Экзюпери, Год издания: 1943
Текущий радиус: 3
Новый радиус: 6
```

## Исходный код программы

#Задание 1

```
class Book:
    def __init__(self, title, author, year):
        self.title = title
        self.author = author
        self.year = year
    def get_info(self):
        return f'Название книги: {self.title}, Автор: {self.author}, Год издания: {self.year}'
```

```
book = Book('Маленький принц', 'Антуан де Сент-Экзюпери', '1943')
print(book.get_info())
```

#Задание 2

```
class Circle:
```

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

def get_radius(self):
    return self.radius

def set_radius(self, new_radius):
    self.radius = new_radius


circle = Circle(3)
print('Текущий радиус:', circle.get_radius())
circle.set_radius(6)
print('Новый радиус:', circle.get_radius())
```

### **Заключение**

В ходе выполнения лабораторной работы были успешно решены следующие задачи:

1. Получен практический опыт работы с ООП в Python.
2. Ознакомился с базовыми классами и методами
3. Выполнена работа с конструктором