

Министерство цифрового развития и массовых коммуникаций  
Российской Федерации

Ордена Трудового Красного Знамени федеральное государственное  
бюджетное образовательное учреждение  
высшего образования  
«Московский технический университет связи и информатики»  
(МТУСИ)

Кафедра Информационных технологий

**Отчет по лабораторной работе № 5**  
по дисциплине «Введение в информационные технологии»  
на тему: «Работа с классами»

Выполнил: студент группы БВТ2402  
Левая Валерия Валерьевна

Москва 2024

**Цель работы:** Получить практический опыт работы с ООП в Python.

**Ход работы:**

**Задание 1:** Базовый класс и методы.

1. Определим класс **Book**, который имеет три атрибута: **title** (название), **author** (автор), и **year** (год издания).
2. Добавим метод `get_info()`, который возвращает информацию о книге в формате: "Название книги: [**title**], Автор: [**author**], Год издания: [**year**]".

```
task1.py X
task1.py > Book
1 class Book:
2
3     def __init__(self, title, author, year):
4         self.title = title
5         self.author = author
6         self.year = year
7
8     def get_info(self):
9         return 'Название книги: {}, Автор: {}, Год издания: {}'.format(self.title, self.author, self.year)
10
11
12 my_book = Book('Мартин Иден', 'Джек Лондон', '1909')
13 print(my_book.get_info())
14
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
● PS C:\Users\levaa\OneDrive\Рабочий стол\ввикт\лаб5> & C:/Users/levaa/AppData/Local
65/task1.py"
Название книги: Мартин Иден, Автор: Джек Лондон, Год издания: 1909
○ PS C:\Users\levaa\OneDrive\Рабочий стол\ввикт\лаб5>
```

**Задание 2:** Работа с конструктором.

1. Определим класс **Circle** для представления круга.
2. Используем конструктор `__init__` для инициализации радиуса круга (**radius**).
3. Добавим метод `get_radius()`, который возвращает значение радиуса.
4. Добавим метод `set_radius(new_radius)`, который позволяет изменить радиус круга.
5. Создадим объект класса **Circle**, измените его радиус и выведите новый радиус на экран

```
task2.py X
task2.py > Circle
1 class Circle:
2     def __init__(self, radius):
3         self.radius = radius
4
5     def get_radius(self):
6         return self.radius
7
8     def set_radius(self, new_radius):
9         self.radius = new_radius
10
11 circle = Circle(10)
12 print('Начальный радиус:', circle.get_radius())
13
14 circle.set_radius(8)
15 print('Новый радиус:', circle.get_radius())
16
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
● PS C:\Users\levaa\OneDrive\Рабочий стол\ввикт\лаб5> & C:
65/task2.py"
Начальный радиус: 10
Новый радиус: 8
○ PS C:\Users\levaa\OneDrive\Рабочий стол\ввикт\лаб5>
```

Вывод: Я получила практический опыт работы с ООП в Python.