

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ



МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

*Факультет Информационных технологий  
Кафедра Информатики и информационных технологий*

направление подготовки

09.03.02 «Информационные системы и технологии»

## ЛАБОРАТОРНАЯ РАБОТА № 1

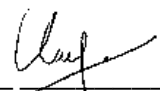
Дисциплина: Объектно-ориентированное программирование

Тема: Создание программы с использованием классов

Выполнил: студент группы 231-338

*Шаура Илья Максимович*  
(Фамилия И.О.)

Дата, подпись 09.09.2024  
(Дата)

  
(Подпись)

Проверил: \_\_\_\_\_  
(Фамилия И.О., степень, звание)

(Оценка)

Дата, подпись \_\_\_\_\_  
(Дата)

(Подпись)

Замечания:

---

---

---

Москва

2024

**Цель:** получить практические навыки в создании программ, содержащих определения классов и использующих объекты на языке программирования C#.

**Обоснование:** при вызове конструкторов, первым всегда вызывается самый последний в цепи вызовов конструктор, так как вызовы складываются в т.н. *стек памяти*, который работает по принципу «первый зашёл – последний вышел».

Поля name, age и email после выполнения следующего кода будут иметь заданные им по умолчанию значения при создании экземпляра объекта Person через конструкторы без передачи соответствующих данных, и, в другом случае, с переданными в конструктор данными.

**Задание.** Создание классов, реализация методов, объявление классов, запуск программы и вывод полученного на экран.

Листинг кода:

```
Person John = new("John"); // Вызов конструктора с 1 аргументом имени
Person Will = new("Will", 20); // Вызов конструктора с 2 аргументами имени и
// возраста
Person Bob = new("", 21, "bob2000@gmail.com"); // Вызов конструктора с 3
// аргументами (имя по умолчанию = Will, возраст и email)

Console.WriteLine("----- John -----");

Console.WriteLine(John.name); // John
Console.WriteLine(John.age); // 18 (default)
Console.WriteLine(John.email); // ben@gmail.com (default)

Console.WriteLine("----- Will -----");

Console.WriteLine(Will.name); // Will
Console.WriteLine(Will.age); // 20
Console.WriteLine(Will.email); // ben@gmail.com (default)

Console.WriteLine("----- Bob -----");

Console.WriteLine(Bob.name); // Bob (default)
Console.WriteLine(Bob.age); // 21
Console.WriteLine(Bob.email); // bob2000@gmail.com
```

Консольный вывод:

```
----- John -----  
John  
18  
ben@gmail.com  
----- Will -----  
Will  
20  
ben@gmail.com  
----- Bob -----  
Bob  
21  
bob2000@gmail.com
```

Описанный в задании класс:

```
class Person  
{  
    public string name = "Ben";  
    public int age = 18;  
    public string email = "ben@gmail.com";  
  
    public Person(string name)  
    {  
        this.name = name;  
    }  
  
    public Person(string name, int age) : this(name)  
    {  
        this.age = age;  
    }  
  
    public Person(string name, int age, string email) : this("Bob", age)  
    {  
        this.email = email;  
    }  
}
```