МИНЕСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное учреждение высшего образования

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

Лабораторная работа №7

**Работа с классами**

**по дисциплине**

**«Введение в информационные технологии»**

Выполнил: студент гр. БВТ2403

Редько Е.В.

Проверил:

Москва, 2024 г.

**Цель работы:** разработать систему управления сотрудниками, демонстрирующую множественное наследование, инкапсуляцию и полиморфизм в Python. Система должна уметь обрабатывать различные типы сотрудников, включая менеджеров и технических специалистов, а также предоставлять возможность для расширения и добавления новых ролей.

**Оборудование:** язык программирования Phyton.

**Практика:**

Задание 1. Создадим класс Employee с атрибутами name, id и методом ger\_info

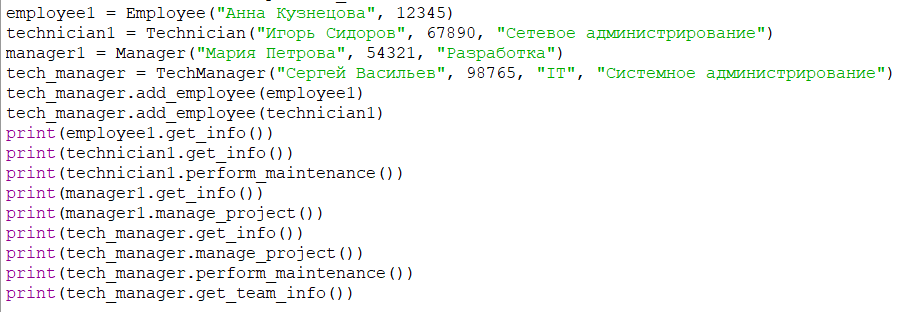
Задание 2. Создадим класс Manager, наследующий от Employee, с дополнительным атрибутом department и методами get\_info() и manage\_project()

Задание 3. Создадим класс Technician, наследующий от Employee, с уникальным атрибутом specialization и методами perform\_maintenance() и get\_info().

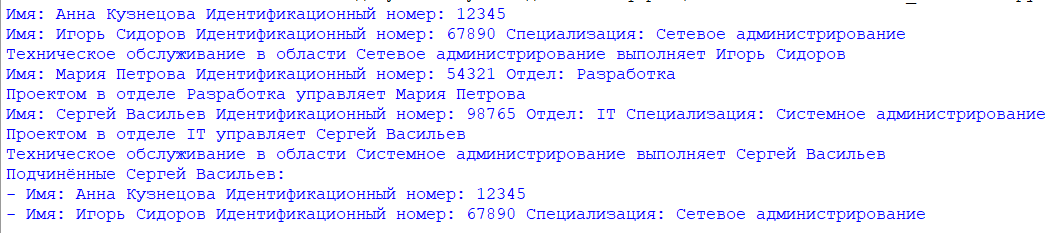
Задание 4. Также, создадим класс Tech\_Manager, наследующий от Manager и Technician, с методами manage\_project(), perform\_maintenance(), get\_info(), add\_employee(), который позволяет TechManager добавлять сотрудников в список подчинённых, и get\_team\_info().



Создадим объекты каждого класса и продемонстрируем их функциональность.



Результат работы программы:



**Вывод:** мы получили практический опыт работы с ООП в Python.