

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 № 7
по дисциплине «Введение в информационные технологии»
на тему: «Работа с классами»

Выполнил: студент группы БВТ2402
Левая Валерия Валерьевна

Москва 2024

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.

Ход работы:

Задание:

1. Создадим класс **Employee** с общими атрибутами, такими как **name** (имя), **id** (идентификационный номер) и методами, например, **get_info()**, который возвращает базовую информацию о сотруднике.
2. Создадим класс **Manager** с дополнительными атрибутами, такими как **department** (отдел) и методами, например, **manage_project()**, символизирующим управление проектами.
3. Создадим класс **Technician** с уникальными атрибутами, такими как **specialization** (специализация), и методами, например, **perform_maintenance()**, означающим выполнение технического обслуживания.
4. Создадим класс **TechManager**, который наследует как **Manager**, так и **Technician**. Этот класс должен комбинировать управленческие способности и технические навыки, например, иметь методы для управления проектами и выполнения технического обслуживания.
5. Добавим метод **add_employee()**, который позволяет **TechManager** добавлять сотрудников в список подчинённых.
6. Реализуем метод **get_team_info()**, который выводит информацию о всех подчинённых сотрудниках.
7. Создадим объекты каждого класса и продемонстрируем их функциональность.

Код:

```
done.py x
done.py > Manager > manage_project > project_name

1 class Employee:
2     def __init__(self, name, id):
3         self.name = name
4         self.id = id
5
6     def get_info(self):
7         return f'Имя сотрудника: {self.name}, его идентификационный номер: {self.id}'
8
9
10 class Manager(Employee):
11     def __init__(self, name, id, department):
12         Employee.__init__(self, name, id)
13         self.department = department
14
15     def manage_project(self, project_name):
16         return f'Сотрудник {self.name} управляет проектом {project_name} в {self.department} в отделе'
17
18
19 class Technician(Employee):
20     def __init__(self, name, id, specialization):
21         Employee.__init__(self, name, id)
22         self.specialization = specialization
23
24     def perform_maintenance(self, equipment):
25         return f'Сотрудник {self.name} со специализацией {self.specialization} выполняет техническое обслуживание {equipment}'
26
27 class TechManager(Manager, Technician):
28     def __init__(self, name, id, department, specialization):
29         Manager.__init__(self, name, id, department)
30         Technician.__init__(self, name, id, specialization)
31         self.team = []
32
33     def add_employee(self, employee):
34         self.team.append(employee)
35
36     def get_team_info(self):
37         if not self.team:
38             return f'У {self.name} еще нет подчиненных'
39         team_info = [employee.get_info() for employee in self.team]
40         return f'Команда {self.name}: \n' + '\n'.join(team_info)
41
42
43 manager = Manager('Егор', 2, 'Акции')
44 technician = Technician('Матвей', 3, 'Электрика')
45 tech_manager = TechManager('Владос', 4, 'Продажи', 'Механик')
46
47 print(employee.get_info())
48 print(manager.get_info())
49 print(technician.get_info())
50 print(tech_manager.get_info())
51
52 print(manager.manage_project('Увеличение продаж'))
53 print(technician.perform_maintenance('Компьютера'))
54
55 tech_manager.add_employee(employee)
56 tech_manager.add_employee(manager)
57 tech_manager.add_employee(technician)
58
59 print(tech_manager.get_team_info())
60
61
62
63
64
65
66
67
68
```

Итого:

```
& C:/Users/levaa/AppData/Local/Programs/Python/Python312/python.exe "c:/U
й стол\ввикт\l7\done.py"
Имя сотрудника: Икром, его идентификационный номер: 1
Имя сотрудника: Егор, его идентификационный номер: 2
Имя сотрудника: Матвей, его идентификационный номер: 3
Имя сотрудника: Владос, его идентификационный номер: 4
Сотрудник Егор управляет проектом Увеличение продаж в Акции в отделе)
Сотрудник Матвей со специализацией Электрика выполняет техническое обслуживание Компьютера
Команда Владос:
Имя сотрудника: Икром, его идентификационный номер: 1
Имя сотрудника: Егор, его идентификационный номер: 2
Имя сотрудника: Матвей, его идентификационный номер: 3
PS C:\Users\levaa\OneDrive\Рабочий стол\ввикт\l7>
```