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 «Интеллектуальные системы в управлении и автоматизации»

ЛАБОРАТОРНАЯ РАБОТА № 1

«Работа со списками Python.»

ПО ДИСЦИПЛИНЕ

«Введение в объектно-ориентированное программирование»

Выполнил:

студент 2-го курса

группы БАП2301

Скрипников Андрей

Проверила:

Хохлова Мария Владимировна

# Москва 2024

**Цель работы**

Ознакомиться с основами объектно-ориентированного программирования (ООП) в Python, получить навыки разработки объектно-ориентированного кода.

**Задачи работы:**

1. изучить лекционный материал на тему Введение в объектно-ориентированное программирование в Python;
2. выполнить задание для получения навыков разработки объектно-ориентированного кода, реализовывая принцип инкапсуляции, полиморфизма и наследования.
3. оформить отчет о проделанной работе.

**Основные классы программы**

1. Component

- Описывает компонент печатной платы.

- Содержит:

- Атрибут `name` — название компонента.

- Метод `\_\_str\_\_` возвращает название компонента.

2. `Worker`

- Описывает работника.

- Содержит:

- Атрибуты `name` и `role` для имени и роли работника.

- Метод `\_\_str\_\_` возвращает информацию о работнике в формате: `<Имя> (<Роль>)`.

3. `Station`

- Базовый класс для рабочих станций.

- Содержит:

- Атрибуты:

- `name` — название станции.

- `equipment` — список оборудования.

- `workers` — список работников.

- `materials` — список материалов (опционально).

- Метод `operate` реализуется в дочерних классах для выполнения операций.

4. Дочерние классы `Station`

- `AssemblyStation`

- Моделирует сборочную станцию.

- Метод `operate` отмечает плату как собранную.

- `TestingStation`

- Моделирует станцию тестирования.

- Метод `operate` проверяет, собрана ли плата, и тестирует ее.

- `PackagingStation`

- Моделирует станцию упаковки.

- Метод `operate` проверяет, прошла ли плата тестирование, и упаковывает ее.

5. `CircuitBoard`

- Описывает печатную плату.

- Содержит:

- Атрибуты:

- `size` — размер платы.

- `components` — список компонентов.

- Флаги `assembled`, `tested`, `packaged` для состояния платы.

- Метод `\_\_str\_\_` возвращает информацию о плате и ее компонентах.

Производственный процесс

Производственный процесс организован функцией `manufacture\_circuit\_board`, которая:

1. Создает объект платы `CircuitBoard`.

2. Последовательно вызывает метод `operate` для станций:

- Сборка на `AssemblyStation`.

- Тестирование на `TestingStation`.

- Упаковка на `PackagingStation`.

3. Проверяет состояние платы и выводит результаты.

Пример выполнения

Входные данные:

- Размер платы: "маленький".

- Компоненты: `["микросхема", "резисторы", "диоды", "провода"]`.

**Заключение**

Код успешно моделирует производственный процесс, включает проверку логики операций (например, невозможность тестирования не собранной платы) и обеспечивает четкую последовательность действий.