Лабораторная №4

1. Информация о паттерне "Прототип" и его тип

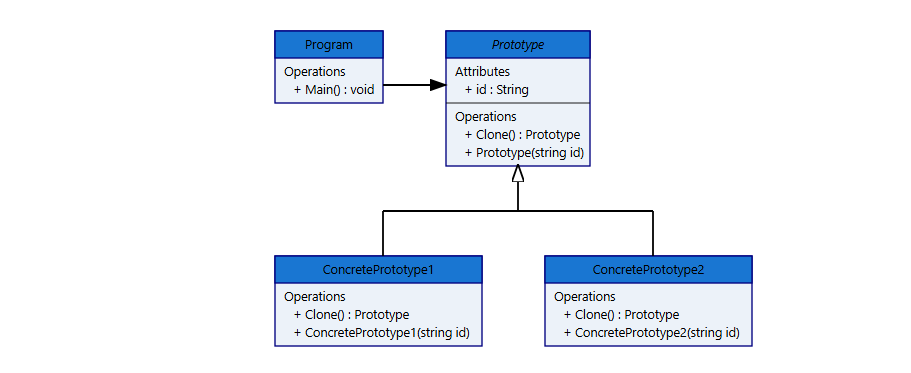
Паттерн "Прототип" — это порождающий паттерн проектирования, который позволяет копировать объекты, не прибегая к их созданию заново. Вместо этого используется метод клонирования, который создает новые объекты на основе существующих. Он используется для создания копий объектов (прототипов) без необходимости повторной инициализации или обращения к их конструкторам. Это особенно полезно, если создание объектов "с нуля" связано с высокой стоимостью или сложностью.

Основной принцип: создаётся прототип объекта, который затем клонируется для создания новых экземпляров. Прототип поддерживает интерфейс клонирования (например, метод clone()), позволяющий клиенту создавать копии, не зная конкретных деталей реализации.

Паттерн прототип стоит использовать когда создание объектов дорогостоящее: Если создание нового объекта занимает много времени или ресурсов, клонирование существующего объекта может быть более эффективным. Также когда Когда нужно создавать объекты с разными конфигурациями: Легче клонировать прототип и изменить его, чем создавать новый объект с нуля.

Тип паттерна порождающий, значит он отвечает за удобное и безопасное создание новых объектов или даже целых семейств объектов.

**2. UML-диаграмма**



UML-диаграмма для паттерна "Прототип" включает:

* **Prototype**: интерфейс или абстрактный класс с методом clone().
* **ConcretePrototype1**: реализация интерфейса Prototype, определяющая логику клонирования.
* **ConcretePrototype2**: использует метод clone() для создания новых объектов на основе прототипа.

**3. Пример**

* В игре мы создаём один объект дерева, который служит прототипом. Используя метод клонирования, мы создаём множество таких же деревьев с небольшими вариациями. Это экономит память и упрощает процесс генерации игровых уровней.
* Системы управления базами данных: Прототипы могут использоваться для создания шаблонов запросов и объектов базы данных, которые затем клонируются и модифицируются для конкретных нужд.
* Проектирование и моделирование: В проектировании сложных систем, таких как архитектура зданий или машин, Прототипы используются для создания и модификации шаблонов и моделей​.

## **Какие недостатки у паттерна Прототип?**

1. Сложность клонирования: Некоторые объекты могут быть сложными для клонирования, особенно если они содержат ссылки на другие объекты или используют глубокое копирование.
2. Необходимость реализации клонирования: Каждый класс, который может быть клонирован, должен реализовать метод клонирования, что добавляет дополнительную нагрузку на разработку и сопровождение.
3. Проблемы с клонированием состояний: При клонировании объектов, содержащих сложные состояния или зависимости, могут возникнуть проблемы с консистентностью данных и состоянием объекта​
4. Не каждый объект удастся скопировать таким образом, ведь часть его состояния может быть приватной, а значит — недоступной для остального кода программы.