1. Какие типы паттернов бывают?

Виды шаблонов проектирования

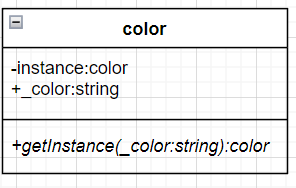
* [Порождающие](https://habrahabr.ru/company/mailru/blog/325492/#porozhdayuschie-shablony-proektirovaniya)
* [Структурные](https://habrahabr.ru/company/mailru/blog/325492/#strukturnye-shablony-proektirovaniya)
* [Поведенческие](https://habrahabr.ru/company/mailru/blog/325492/#povedencheskie-shablony-proektirovaniya)

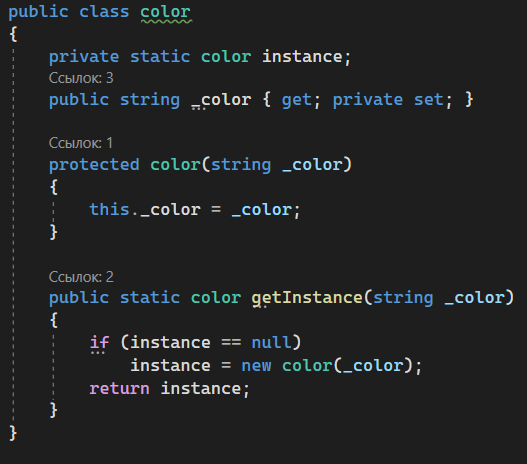
Порождающие шаблоны пытаются решать эти проблемы, управляя способом создания объектов.

Структурными называют шаблоны, которые облегчают проектирование, определяя простой способ реализации взаимоотношений между сущностями.

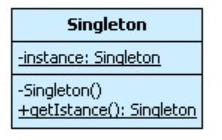
Поведенческие шаблоны проектирования определяют алгоритмы и способы реализации взаимодействия различных объектов и классов.

2. Нарисуете диаграмму классов и поясните принцип работы паттерна Singleton. Назовите условия применения.

**\*Пример из лабораторной работы:**



\*В общем случае:

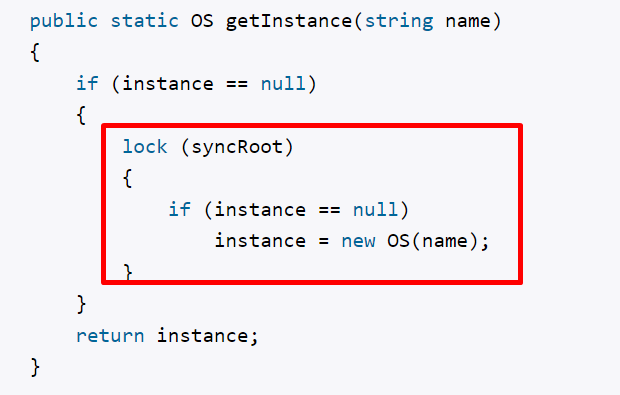


Условия применения: создавать единственный экземпляр некоторого типа, предоставлять к нему доступ извне и запрещать создание нескольких экземпляров того же типа.

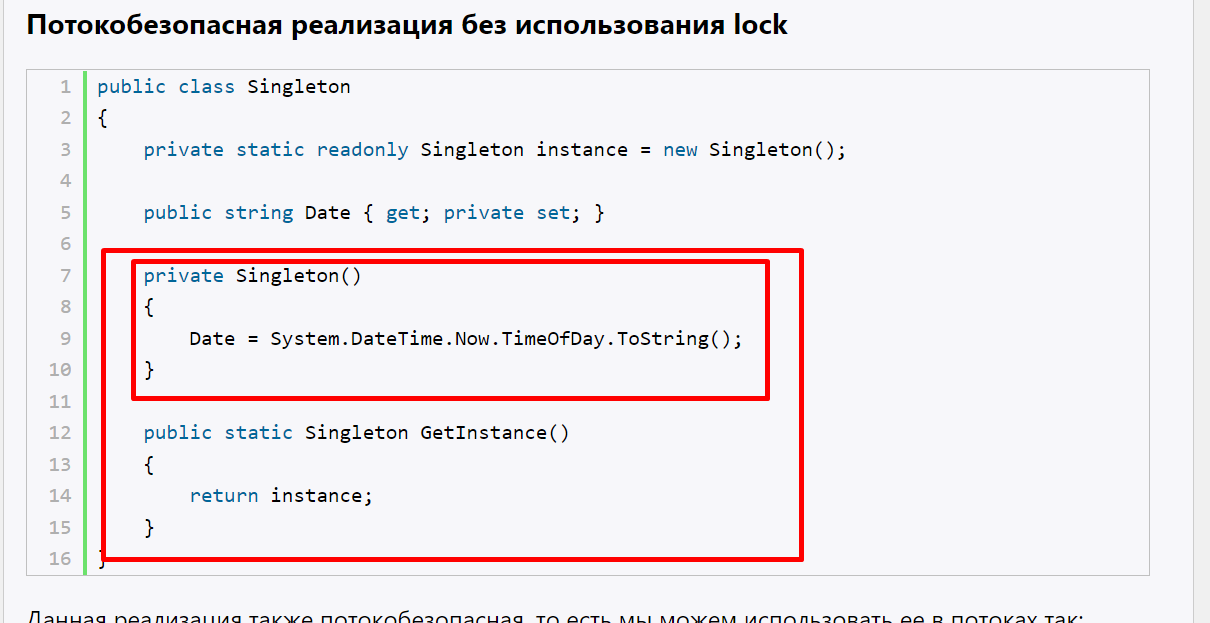
3. Как сделать потокобезопасную, с отложенной реализацией версию Singleton.

При применении паттерна синглтон в многопоточным программах мы можем столкнуться с проблемой доступа с нашему синглтону одновременно с разных потоков, это не позволяет предсказать конечный результат выполнения программы и теряется весь смысл использования данного паттерна.

Чтобы избежать одновременного доступа к коду из разных потоков критическая секция заключается в блок **lock**.

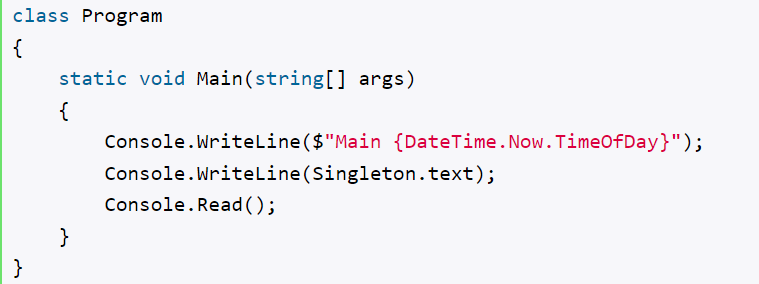


ИЛИ



Определение объекта синглтона в виде статического поля класса открывает нам дорогу к созданию **Lazy-реализации** паттерна Синглтон, то есть такой реализации, где данные будут инициализироваться только перед непосредственным использованием.

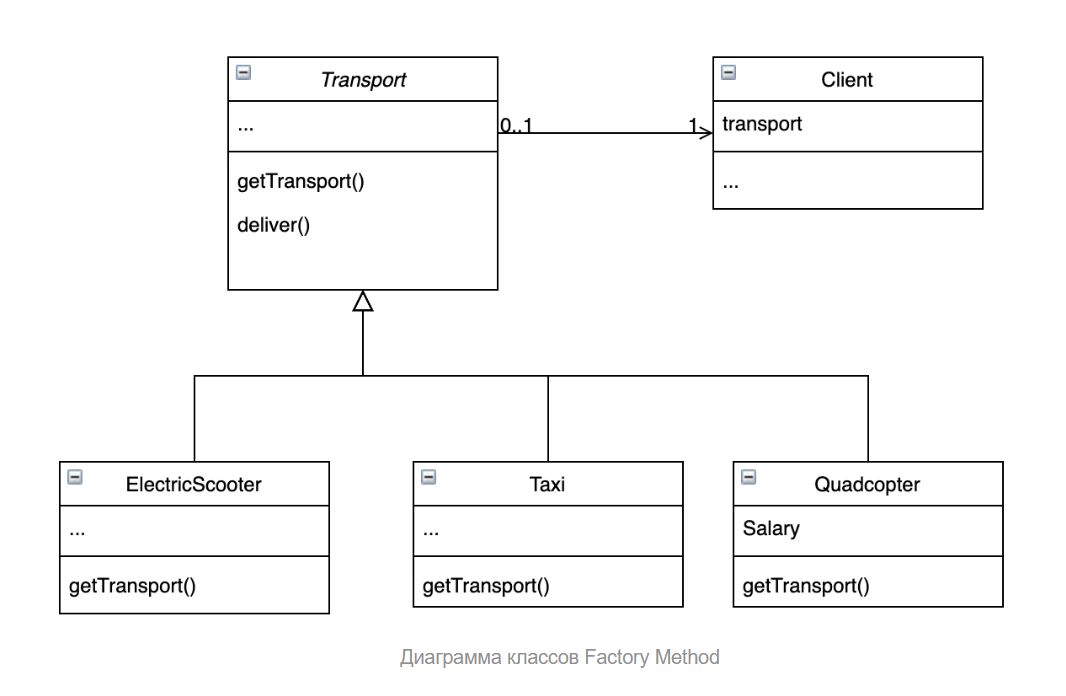




4. Нарисуете диаграмму классов и поясните принцип работы паттерна Factory Method. Назовите условия применения

**Фабричный метод (Factory method)** также известный как Виртуальный конструктор (Virtual Constructor) - пораждающий шаблон проектирования, определяющий общий интерфейс создания объектов в родительском классе и позволяющий изменять создаваемые объекты в дочерних классах.

Посмотрим на диаграмму классов такого подхода.



5. Нарисуете диаграмму классов и поясните принцип работы паттерна Abstract Factory. Назовите условия применения

Паттерн "Абстрактная фабрика" (Abstract Factory) предоставляет интерфейс для создания семейств взаимосвязанных объектов с определенными интерфейсами без указания конкретных типов данных объектов.

Самый первый вопрос, на который нужно ответить самому себе, изучая данный паттерн: «Что же такое Абстрактная фабрика». Самый простой и точный ответ, гласит, что Абстрактная фабрика – это «фабрика фабрик». Но здесь появляется второй вопрос: «Для чего вообще может кому-нибудь понадобиться «фабрика фабрик»?

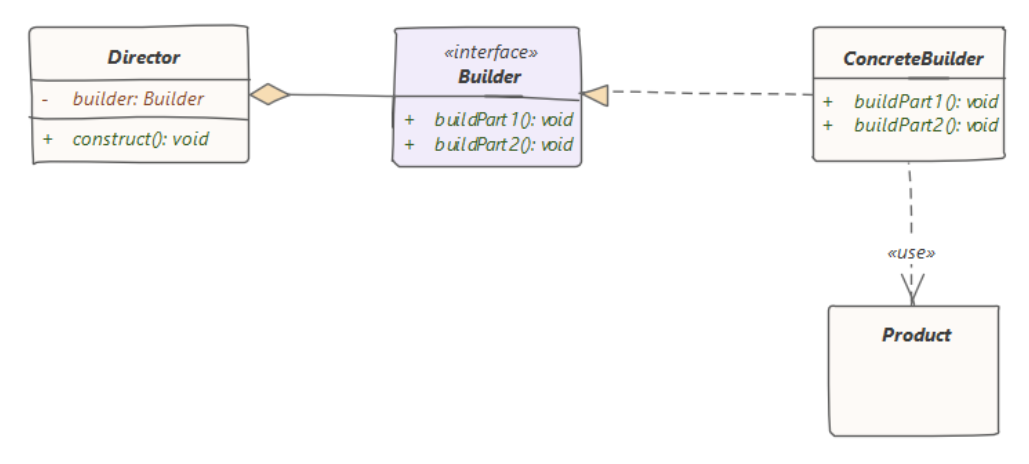


6. Нарисуете диаграмму классов и поясните принцип работы паттерна Prototype. Назовите условия применения

Паттерн Прототип (Prototype) позволяет создавать объекты на основе уже ранее созданных объектов-прототипов. То есть по сути данный паттерн предлагает технику клонирования объектов.

7. Нарисуете диаграмму классов и поясните принцип работы паттерна Builder. Назовите условия применения

Паттерн Builder — это паттерн проектирования, который позволяет поэтапно создавать сложные объекты с помощью четко определенной последовательности действий. Строительство контролируется объектом-распорядителем (director), которому нужно знать только тип создаваемого объекта.



8. Что такое Пул объектов (Object pool), в чем суть паттерна?

[Пул объектов](http://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D1%8B%D0%B9_%D0%BF%D1%83%D0%BB) — порождающий шаблон проектирования, набор инициализированных и готовых к использованию объектов.

Пул объектов предназначен хранения готовых к использованию объектов. Когда системе требуется новый объект, он запрашивается из Пула, минуя процесс порождения. А после использования возвращается обратно в Пул вместо уничтожения.

Шаблон применяется для повышения производительности, если:

* объекты часто создаются и уничтожаются;
* в системе существует ограниченное количество объектов типа, хранимого в Пуле;
* создание и/или уничтожение объекта являются очень затратными операциями.

Пул объектов может работать как с интерфейсами, так и с конкретными реализациями. Все зависит от архитектуры разрабатываемой системы и решаемых задач.

9. Какое основное назначение паттерна отложенная инициализация (Lazy initialization)?

Отложенная инициализация или «ленивая» инициализация — это способ доступа к объекту, скрывающий за собой механизм, позволяющий отложить создание этого объекта до момента первого обращения. Необходимость ленивой инициализации может возникнуть по разным причинам: начиная от желания снизить нагрузку при старте приложения и заканчивая оптимизацией редко используемого функционала. И действительно, не все функции приложения используются всегда и, тем более, сразу, потому создание объектов, реализующих их, вполне рационально отложить до лучших времён.