**Курс:** Шаблоны проектирования приложений

**Тема:** Модуль 10 Структурные паттерны. Фасад. Компоновщик

**Цель:**

Освоить применение структурного паттерна "Фасад", который предоставляет упрощённый интерфейс к сложной системе. В результате выполнения задания вы научитесь проектировать класс-фасад для удобной работы с подсистемами.

**Задание:**

Вам предстоит создать систему управления мультимедиа центром, состоящим из различных подсистем: телевизора, аудиосистемы, DVD-проигрывателя и игровой консоли. Необходимо реализовать паттерн "Фасад", который будет предоставлять простой интерфейс для включения и выключения всей системы, а также выполнения базовых сценариев (например, начать просмотр фильма или запуск видеоигры).

**Описание задания:**

1. Реализуйте классы для каждой подсистемы:
   * **Телевизор (TV)**: Включение, выключение и выбор канала.
   * **Аудиосистема (AudioSystem)**: Включение, выключение, регулировка громкости.
   * **DVD-проигрыватель (DVDPlayer)**: Воспроизведение, пауза, остановка.
   * **Игровая консоль (GameConsole)**: Включение, запуск игры.
2. Создайте класс **HomeTheaterFacade**, который объединяет управление всеми подсистемами и предоставляет упрощённые методы для:
   * Включения всей системы для просмотра фильма.
   * Выключения системы.
   * Запуска игровой консоли.

**Структура работы:**

**1. Подсистемы мультимедийного центра:**

* **TV**: Включение/выключение, выбор канала.
* **AudioSystem**: Включение/выключение, настройка громкости.
* **DVDPlayer**: Воспроизведение/пауза/остановка DVD-диска.
* **GameConsole**: Включение консоли, запуск игры.

**2. Фасад HomeTheaterFacade: Класс, который предоставляет простой интерфейс для управления мультимедийной системой.**

**Задания:**

1. Реализуйте предложенные классы для управления подсистемами мультимедиа.
2. Реализуйте фасад **HomeTheaterFacade**, который позволяет управлять всеми устройствами через простой интерфейс.
3. Добавьте дополнительные сценарии, например:
   * Включение всей системы для прослушивания музыки (включить TV и аудиосистему, установить аудиовход на TV).
   * Возможность регулировки громкости через фасад.
4. Напишите клиентский код, который использует фасад для управления всей системой в различных сценариях (просмотр фильма, запуск игры и т.д.).

**Цель:**

Освоить применение структурного паттерна "Компоновщик" (Composite), который позволяет работать с иерархией объектов единообразно. В результате выполнения задания вы научитесь проектировать системы, в которых как отдельные объекты, так и группы объектов могут обрабатываться одинаково.

**Задание:**

Создать систему управления файлами и папками, которая позволит работать с файлами и папками одинаковым образом. Каждая папка может содержать как файлы, так и другие папки (вложенные), а файлы не могут содержать другие объекты. Задача — реализовать систему с использованием паттерна "Компоновщик".

**Описание задания:**

1. **Иерархия файловой системы**:
   * **Файл (File)**: Имеет имя и размер.
   * **Папка (Directory)**: Может содержать файлы и другие папки. Также имеет имя.
2. **Компонент**:
   * Создать общий интерфейс или абстрактный класс **FileSystemComponent**, который будет содержать общие методы для файлов и папок. Он должен иметь такие методы, как:
     + Display() — для вывода информации о компоненте.
     + GetSize() — для получения размера компонента.
3. **Классы**:
   * **File**: Класс, который реализует компонент, представляющий собой файл.
   * **Directory**: Класс, который реализует компонент, представляющий собой папку, которая может содержать другие компоненты (файлы и папки).
4. **Клиентский код**:
   * Создать иерархию папок и файлов с помощью созданных классов. Вывести информацию о содержимом папок и их размере.

**Задания:**

1. Реализуйте интерфейс или абстрактный класс **FileSystemComponent** с методами для вывода информации о компоненте и получения его размера.
2. Создайте классы **File** и **Directory** для файлов и папок соответственно.
3. Реализуйте возможность добавления и удаления файлов и папок в классе **Directory**.
4. Напишите клиентский код, который создаёт иерархию файлов и папок, выводит её содержимое и рассчитывает общий размер.
5. Добавьте в **Directory** проверку на наличие компонента перед добавлением или удалением.