

1. Клас ApplicationStarter (Стартер на приложението)

Атрибути:

Класът няма член-променливи (атрибути). Целта му е да стартира приложението и да управлява взаимодействията с потребителя чрез менюто.

Методи:

- **Конструктор и Деструктор:**
 - `ApplicationStarter()`: Конструктор без параметри. Инициализира обекта на класа.
 - `~ApplicationStarter()`: Деструктор. Не се използва за специално освобождаване на ресурси в този клас.
- **Основни методи:**
 - `void testInput(OrderSystem& orderSystem)`: Този метод е помощен и служи за добавяне на тестови данни (например доставчици и оптики) в системата, като инициализира няколко примера за доставчици и оптики в `orderSystem`.
 - `void mainMenu(Handler<InputHandler>& ioHandler, OrderSystem& orderSystem)`: Този метод е основното меню на приложението. Той показва на потребителя различни опции, като добавяне на доставчик, добавяне на оптика към доставчик и поставяне на поръчка. В зависимост от избора на потребителя, извиква съответните методи на класа `InputHandler` или `OrderSystem`.
 - `void start()`: Този метод е основната точка на стартиране на приложението. Той инициализира обекти от класовете `InputHandler` и `OrderSystem` и след това извиква метода `mainMenu()`, който осигурява взаимодействие с потребителя чрез конзолата.

2. Клас Deliverer (Доставчик)

Атрибути:

- `std::string bulstat`: Булстат номер на доставчика.
- `std::string name`: Име на доставчика.
- `std::string location`: Локация на доставчика.
- `std::string phoneNumber`: Телефонен номер на доставчика.
- `std::vector<Optics> opticsInStorage`: Списък от обекти тип `Optics`, които са на склад при доставчика.

Методи:

- **Конструктори и Деструктор:**
 - Deliverer(): Конструктор без параметри.
 - Deliverer(const std::string&, const std::string&, const std::string&, const std::string&, const std::vector<Optics>&): Конструктор с параметри за инициализация на атриутите.
 - ~Deliverer(): Деструктор.
- **Методи за работа с данни:**
 - void displayOptics() const: Извежда информация за оптиките на склад.
 - void addOptics(const Optics&): Добавя нова оптика в склада.
 - std::string getBulstat() const: Връща булстат на доставчика.
 - std::string getName() const: Връща името на доставчика.
 - std::string getLocation() const: Връща локацията на доставчика.
 - std::string getPhoneNumber() const: Връща телефонния номер на доставчика.
 - std::vector<Optics> getOpticsInStorage() const: Връща списък с оптики на склад.
 - void setBulstad(const std::string&): Задава нов булстат на доставчика.
 - void setName(const std::string&): Задава ново име на доставчика.
 - void setLocation(const std::string&): Задава нова локация на доставчика.
 - void setPhoneNumber(const std::string&): Задава нов телефонен номер на доставчика.
 - void setOpticsInStorage(const std::vector<Optics>&): Задава нов списък с оптики на склад.

3. Клас Optics (Оптика)

Атрибути:

- **std::string type:** Тип на оптиката (например "Леща", "Филтър").
- **std::string material:** Материал на оптиката (например "Стъкло", "Пластмаса").
- **float thickness:** Дебелина на оптиката в милиметри.
- **float diopter:** Диоптър на оптиката.
- **float price:** Цена на оптиката.

Методи:

- **Конструктори и Деструктор:**

- `Optics()`: Конструктор без параметри.
- `Optics(const std::string&, const std::string&, const float&, const float&)`: Конструктор с параметри за инициализация на атрибутите.
- `~Optics()`: Деструктор.
- **Методи за работа с данни:**
 - `std::string getType() const`: Връща типа на оптиката.
 - `std::string getMaterial() const`: Връща материала на оптиката.
 - `float getThickness() const`: Връща дебелината на оптиката.
 - `float getDiopter() const`: Връща диоптъра на оптиката.
 - `float getPrice() const`: Връща цената на оптиката.
 - `void setType(const std::string&)`: Задава типа на оптиката.
 - `void setMaterial(const std::string&)`: Задава материала на оптиката.
 - `void setThickness(float)`: Задава дебелината на оптиката.
 - `void setDiopter(float)`: Задава диоптъра на оптиката.
 - `void setPrice(float)`: Задава цената на оптиката.

4. Клас InputHandler (Обработчик на входни данни)

- **Методи:**
- **Конструктор и Деструктор:**
 - `InputHandler()`: Конструктор без параметри.
 - `~InputHandler()`: Деструктор.
- **Методи за входни данни:**
 - `Optics inputNewOptics()`: Въвежда нова оптика от потребителя.
 - `void inputOpticsToDeliverer(OrderSystem&)`: Добавя оптика към доставчик в системата.
 - `void inputNewDeliverer(OrderSystem&)`: Добавя нов доставчик в системата.

5. Клас OrderSystem (Система за поръчки)

Атрибути:

- `const std::string ordersLogFileName`: Път до файл за лог на поръчки.
- `const std::string deliverersLogFileName`: Път до файл за лог на доставчици.
- `std::vector<Deliverer> deliverers`: Списък с доставчици, управявани от системата.

Методи:

- **Конструктори и Деструктор:**
 - OrderSystem(): Конструктор без параметри.
 - OrderSystem(const std::vector<Deliverer>&): Конструктор с параметри за инициализация на списъка с доставчици.
 - ~OrderSystem(): Деструктор.
- **Методи за работа със системата:**
 - void addDeliverer(const Deliverer&): Добавя нов доставчик в системата.
 - void displayDelivereres() const: Извежда информация за всички доставчици.
 - void saveDelivererToFile(const Deliverer&) const: Записва данни за доставчик в лог файл.
 - void saveDeliverersToFile() const: Записва всички доставчици в лог файл.
 - void saveOrderToFile(const Deliverer&, const std::vector<Optics>&, float) const: Записва поръчка в лог файл.
 - Deliverer selectDeliverer() const: Избира доставчик за поръчка.
 - void placeOrder(): Разполага поръчка, като избира доставчик и оптика.
 - std::vector<Deliverer> getDeliverers() const: Връща списък с доставчиците.
 - const std::string getOrdersLogFile() const: Връща пътя до лог файла за поръчки.
 - void setDeliverers(const std::vector<Deliverer>&): Задава нов списък с доставчици.

UML (Universal Modeling Language):

