

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ “ЛЬВІВСЬКА
ПОЛІТЕХНІКА“**

Кафедра систем штучного інтелекту

Лабораторна робота №1

з дисципліни

«Системний аналіз»

Виконав:

Студент групи КН-214

Добрій Назарій

Викладач:

Зімоха.І.О.

Львів – 2021р.

Завдання :

Реалізувати патерн «Стратегія» на основі предметної області.

Стратегія – поведінковий патерн проектування, який визначає сімейство схожих алгоритмів і розміщує кожен з них у власному класі. Після цього алгоритми можна замінити один на інших прямо під час виконання програми.

Кроки реалізації

1. Визначте алгоритм, що схильний до частих змін. Також підійде алгоритм, який має декілька варіацій, які обираються під час виконання програми.

2. Створіть інтерфейс стратегій, що описує цей алгоритм. Він повинен бути спільним для всіх варіантів алгоритму.

3. Помістіть варіації алгоритму до власних класів, які реалізують цей інтерфейс.

4. У класі контексту створіть поле для зберігання посилання на поточний об'єкт стратегію, а також метод для її зміни. Переконайтеся в тому, що контекст працює з цим об'єктом тільки через загальний інтерфейс стратегій.

5. Клієнти контексту мають подавати до нього відповідний об'єкт-стратегію, коли хочуть, щоб контекст поведився певним чином.

Предметна область:

№ вар	Предметна область
5	Видання книг

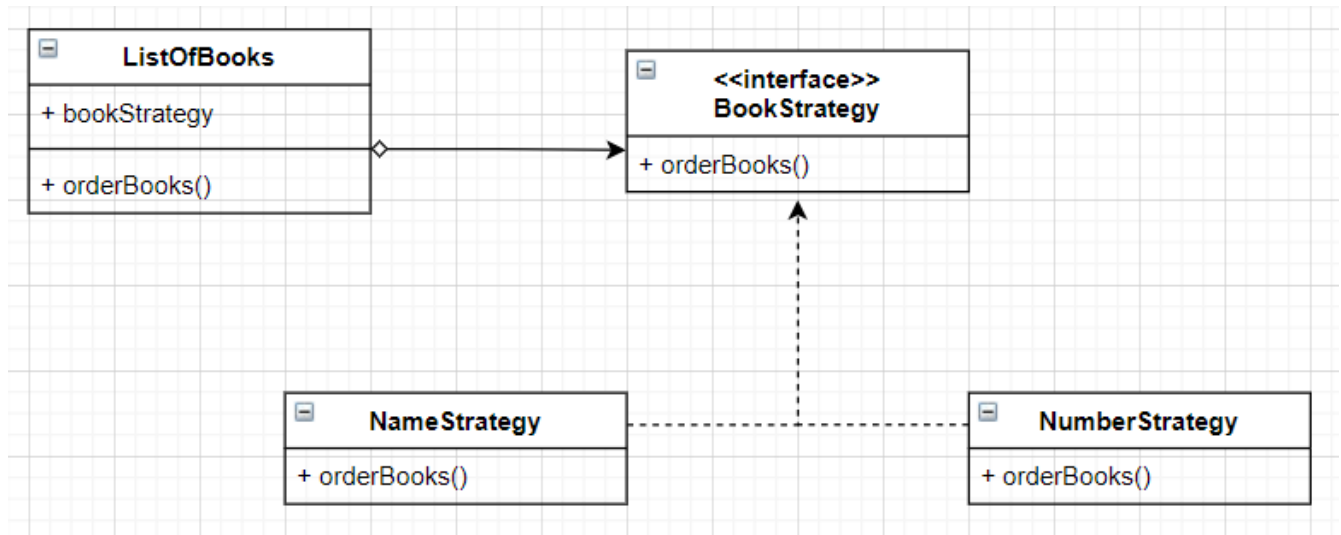


Рис. 1. UML-діаграма патерну “Стратегія”

Код програми на мові Python:

```

from abc import ABC, abstractmethod

class BookStrategy(ABC):
    @abstractmethod
    def order(self, list):
        pass

class OrderByNameStrategy(BookStrategy):
    def order(self, list):
        print("Ordering by client's last name...")
        list.sort(key = lambda x: x.client, reverse=False)
        for e in list:
            e.showOrder()

class OrderByNumberStrategy(BookStrategy):
    def order(self, list):
        print("Ordering by order's number")
        list.sort(key=lambda x: x.number, reverse=False)
        for e in list:
            e.showOrder()

class Order:
    def __init__(self, number, client, date, desc):
        self.number = number
        self.client = client
        self.date = date
        self.desc = desc

    def showOrder(self):
        print('=====')
        print("Номер замовлення: " + str(self.number) )
        print("Клієнт: " + self.client)
        print("Дата: " + self.date)
        print("Опис: " + self.desc)
  
```

```

        print('=====')

class ListOfBooks:
    def __init__(self):
        self.orders = []

    def addOrder(self, order):
        self.orders.append(order)

    def orderOrders(self, strategy: BookStrategy):
        strategy.order(self.orders)

order1 = Order(66, "Назар", "2021-05-12", "опис книги")
order2 = Order(55, "Віталій", "2021-05-15", "опис книги")
order3 = Order(111, "Роман", "2021-05-06", "опис книги")

list = ListOfBooks()

list.addOrder(order1)
list.addOrder(order2)
list.addOrder(order3)

print("Виберіть тип сортування: (1 - за прізвищем Клієнта, 2 - за номером замовлення)")
inp = int(input())
if inp == 1:
    list.orderOrders(OrderByNameStrategy())
if inp == 2:
    list.orderOrders(OrderByNumberStrategy())
else:
    pass

```

Рис. 2.1. Код реалізації програми.

```
Виберіть тип сортування: (1 - за прізвищем Клієнта, 2 - за номером замовлення)
1
Ordering by client's last name...
=====
Номер замовлення: 55
Клієнт: Віталій
Дата: 2021-05-15
Опис: опис книги
=====
=====
Номер замовлення: 66
Клієнт: Назар
Дата: 2021-05-12
Опис: опис книги
=====
=====
Номер замовлення: 111
Клієнт: Роман
Дата: 2021-05-06
Опис: опис книги
=====|

Process finished with exit code 0
```

Рис. 2.2. Сортування за прізвищем клієнта.

```
Виберіть тип сортування: (1 - за прізвищем Клієнта, 2 - за номером замовлення)
2
Ordering by order's number
=====
Номер замовлення: 55
Клієнт: Віталій
Дата: 2021-05-15
Опис: опис книги
=====
=====
Номер замовлення: 66
Клієнт: Назар
Дата: 2021-05-12
Опис: опис книги
=====
=====
Номер замовлення: 111
Клієнт: Роман
Дата: 2021-05-06
Опис: опис книги
=====

Process finished with exit code 0
```

Рис. 2.2. Сортування за номером замовлення клієнта.

Висновок: я навчився реалізовувати поведінковий патерн “Стратегія” у предметній області – видання книг.