Министерство науки и высшего образования РФ  
ФГАОУ ВПО

Национальный исследовательский технологический университет «МИСиС»

Институт Информационных технологий и компьютерных наук (ИТКН)

Кафедра Инфокоммуникационных технологий (ИКТ)

**Отчет по лабораторной работе №3**

по дисциплине «Технологии программирования»

на тему «Поведенческие шаблоны программирования»

Выполнил:  
студент группы БИВТ-22-8

Пронин Семён Михайлович

Москва, 2023

**Цель:** Изучить теоретический материал по поведенческим паттернам программирования, реализовать поведенческие паттерны программирования.

**Задачи:**

1. Реализовать шаблон Стратегия

Паттерн Стратегия (Strategy) представляет шаблон проектирования, который определяет набор алгоритмов, инкапсулирует каждый из них и обеспечивает их взаимозаменяемость. В зависимости от ситуации мы можем легко заменить один используемый алгоритм другим. При этом замена алгоритма происходит независимо от объекта, который использует данный алгоритм.

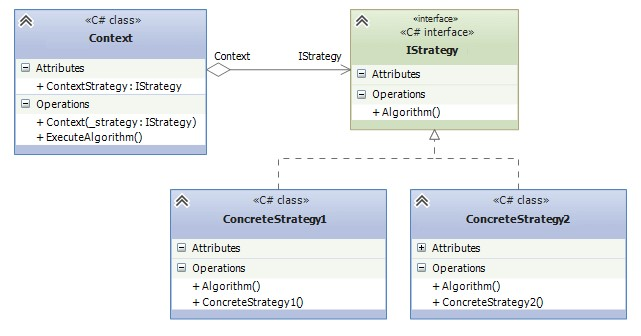


Рис. 1 Паттерн Strategy

1. Реализовать шаблон Цепочка обязанностей

Цепочка Обязанностей (Chain of responsibility) - поведенческий шаблон проектирования, который позволяет избежать жесткой привязки отправителя запроса к получателю. Все возможные обработчики запроса образуют цепочку, а сам запрос перемещается по этой цепочке. Каждый объект в этой цепочке при получении запроса выбирает, либо закончить обработку запроса, либо передать запрос на обработку следующему по цепочке объекту.

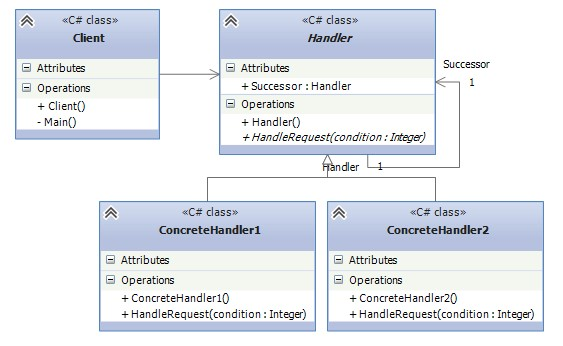


Рис. 2 Паттерн Chain responsibility

1. Реализовать шаблон Итератор

Паттерн Итератор (Iterator) предоставляет абстрактный интерфейс для последовательного доступа ко всем элементам составного объекта без раскрытия его внутренней структуры. Дословно **«Iterator»** можно перевести как **«переборщик»**. То есть это некая сущность, способная перебрать все элементы в коллекции. При этом она позволяет это сделать без вникания во внутреннюю структуру и устройство коллекций.

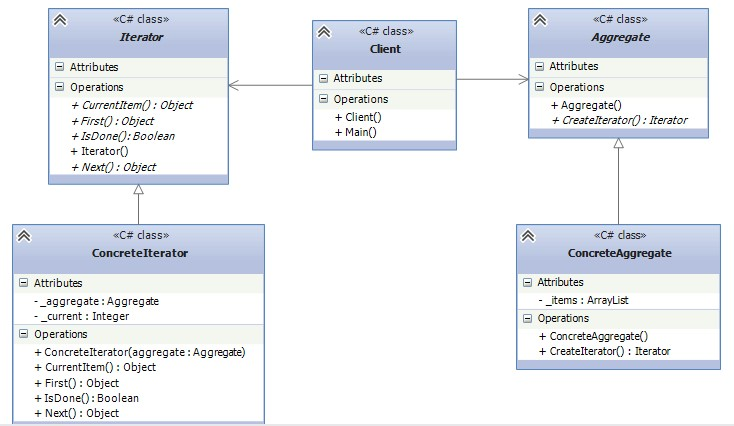


Рис. 3 Паттерн Iterator

**Ход выполнения:**

1. Паттерн Strategy

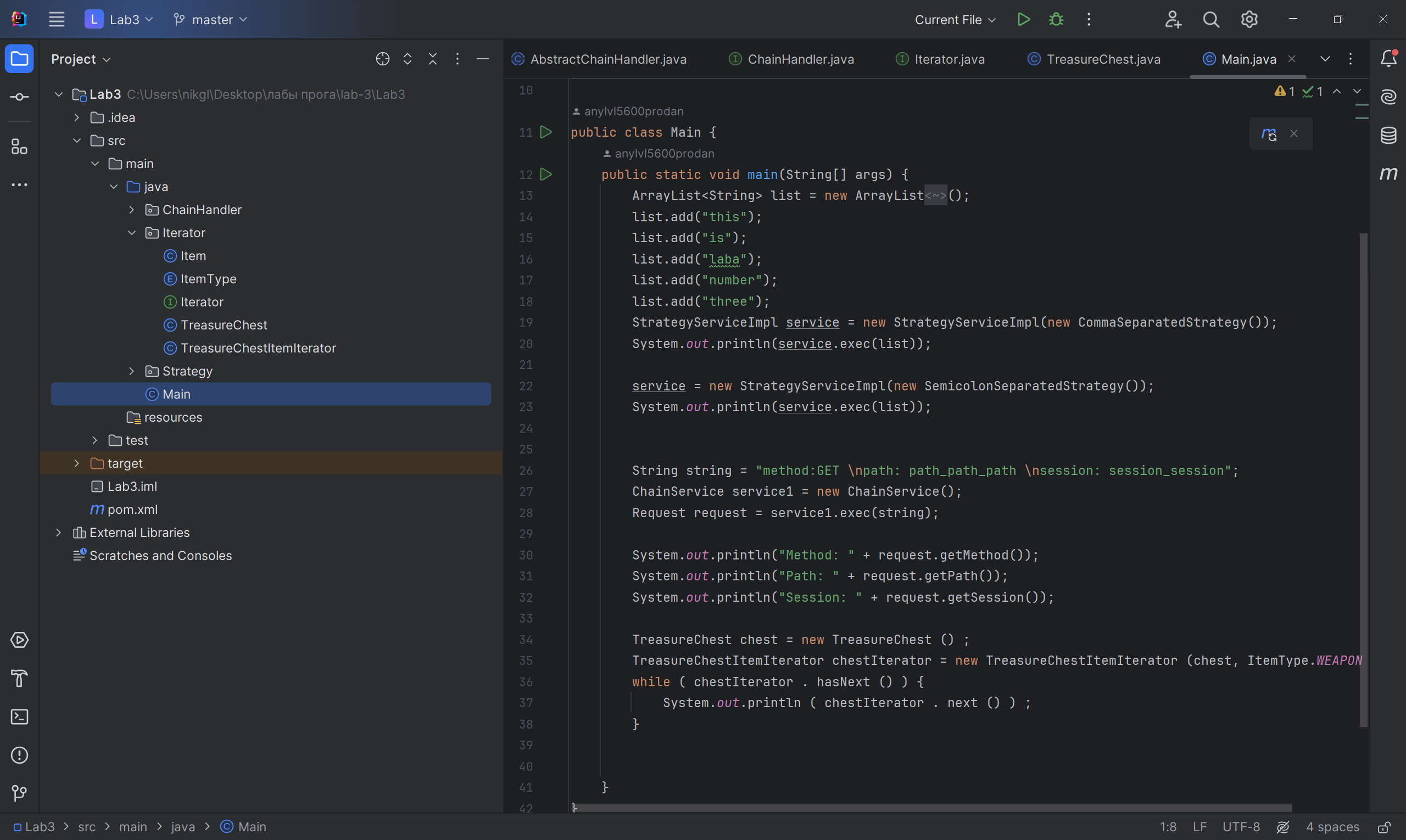


Рис. 5 Реализация паттерна Strategy

1. Паттерн Chain responsibility

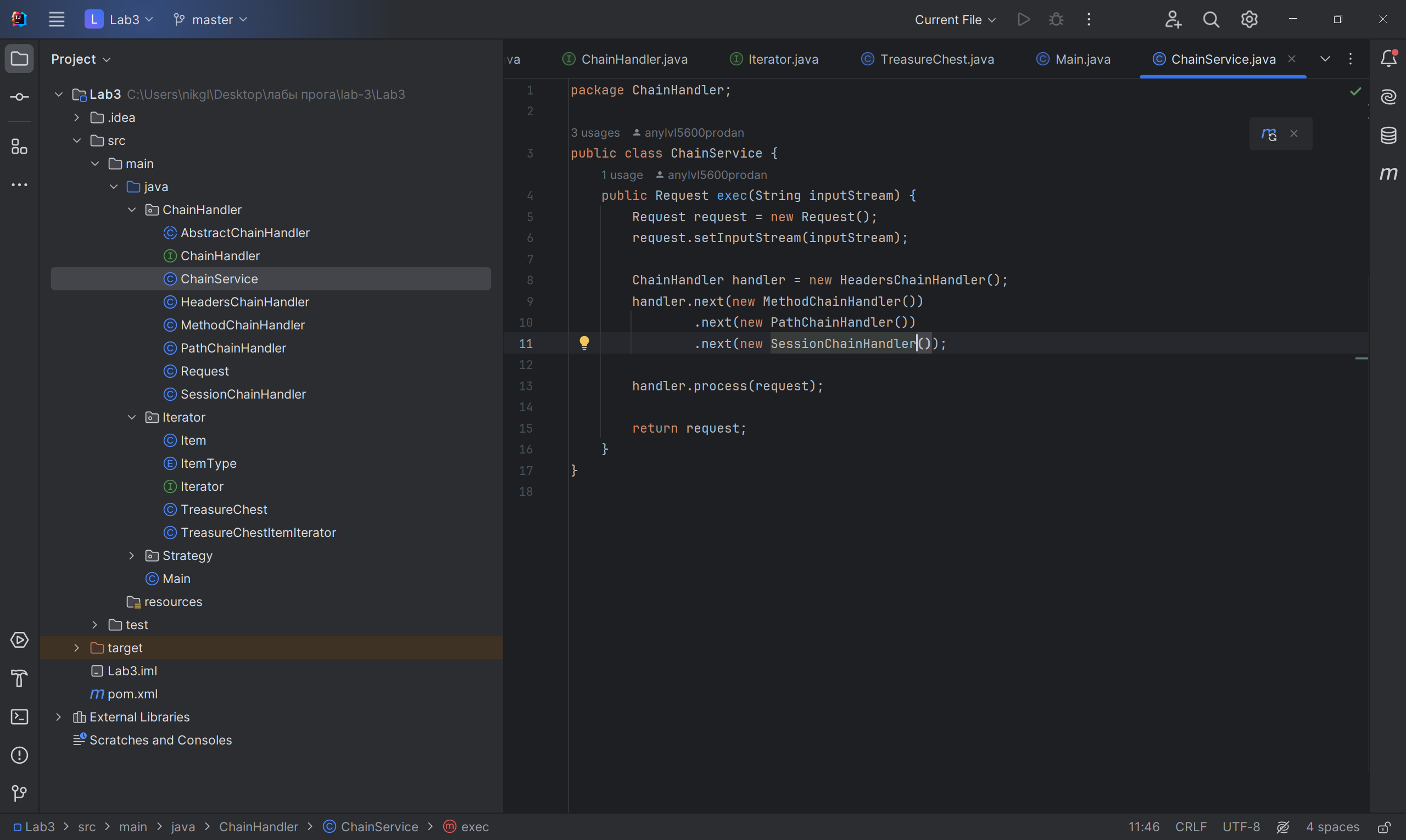


Рис. 6 Реализация паттерна ChainService

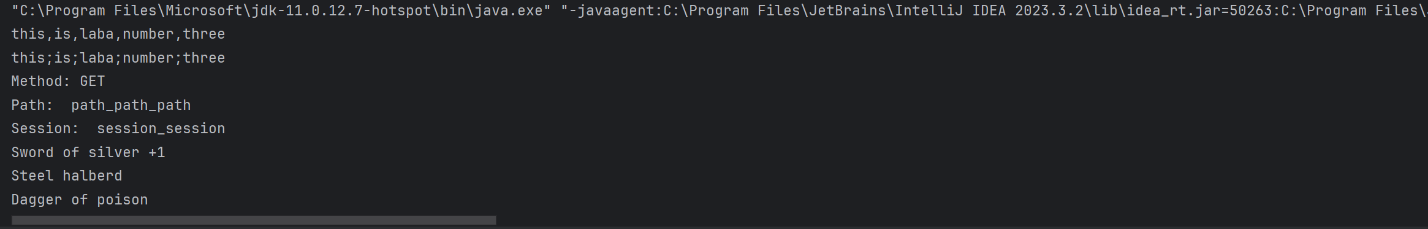


Рис. 7 Вывод программы

1. Паттерн Iterator

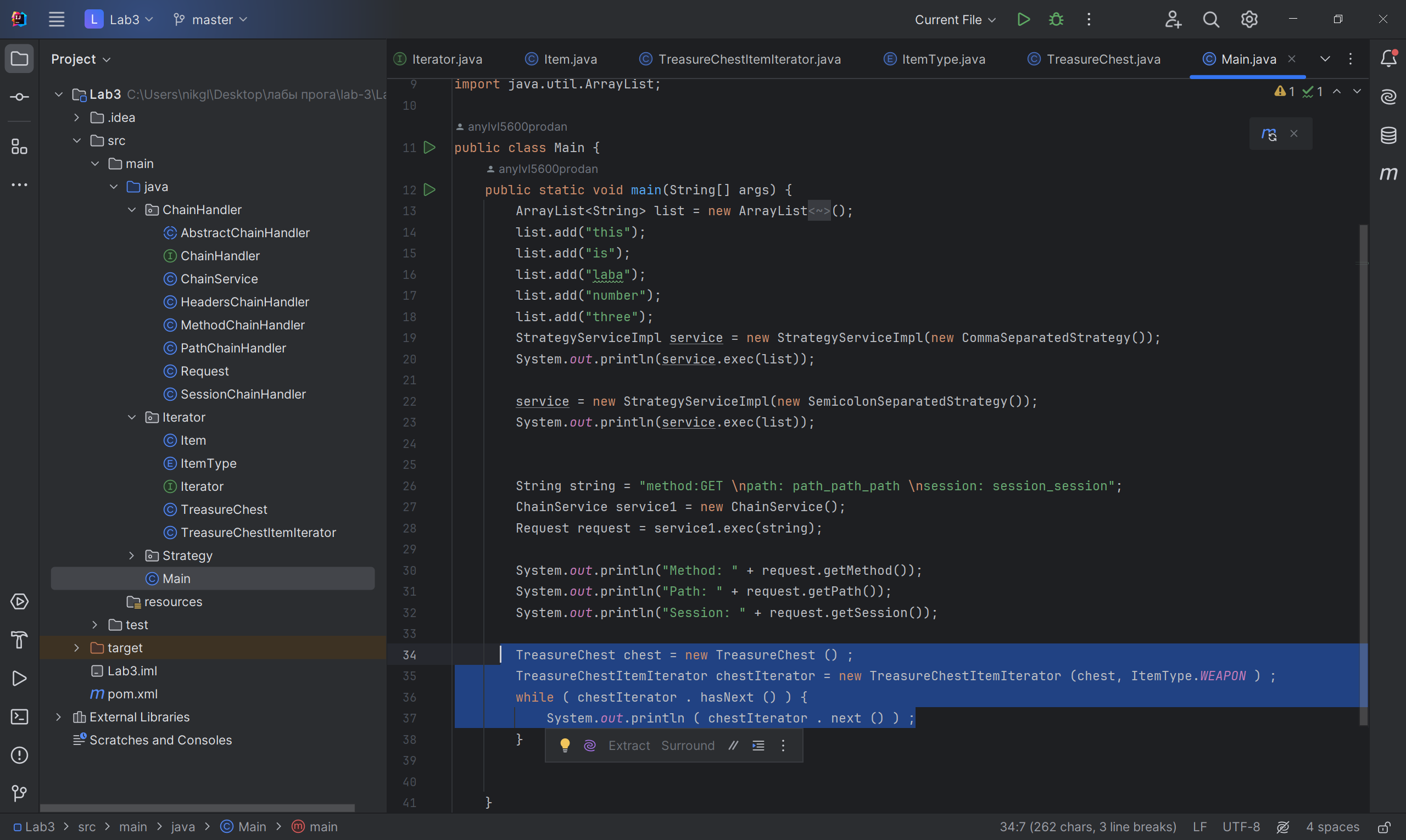


Рис. 8 Реализация паттерна Iterator

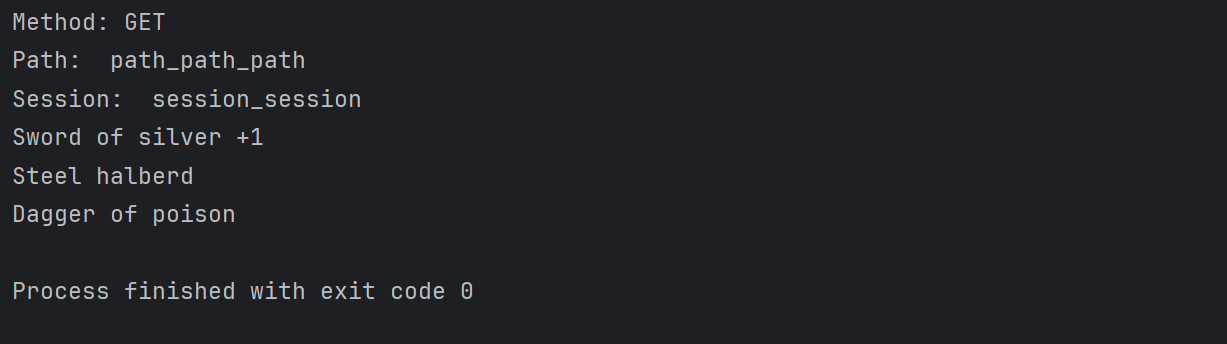


Рис. 9 Вывод программы

**Вывод:** В процессе выполнения данной лабораторной работы, я изучил поведенческие паттерны программирования: Strategy, Chain of responsibility,Iterator. Далее закрепил полученные знания на практике, реализовав паттерны.