

Estevão Cardoso Alves Luzia - 13718745
Leonardo Conde Aoun - 13836149

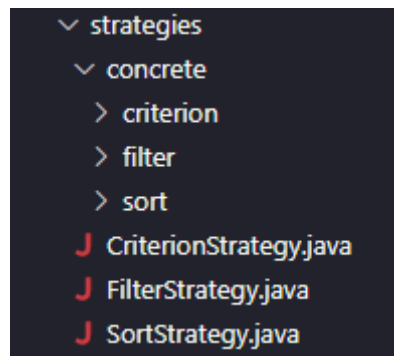
RELATÓRIO - EP2 COO

A primeira modificação no código foi a adição de um diretório chamado “enums” contendo 3 classes enums: “CRIT” (critérios de ordenação), “FILTER” (filtros) e “SORT” (algoritmos de ordenação). A principal vantagem na utilização dessas classes especiais é a melhoria da clareza do código, com nomes significativos e bem definidos para valores constantes usados no decorrer do programa. Outra vantagem interessante é o fortalecimento da segurança de tipos no código, pois os enums oferecem uma verificação em tempo de compilação, não há possibilidade de atribuir um valor que não pertença ao enum, o que pode evitar possíveis erros em tempo de execução.



Posteriormente, o padrão Strategy foi utilizado para diminuir o uso de estruturas condicionais no código, ao invés de diversos blocos condicionais que verificam diferentes variações, o Strategy encapsula cada variação em uma classe separada, permitindo que fique mais fácil adicionar, alterar e remover estratégias sem afetar a lógica principal que está se comunicando com as estratégias no código.

O diretório “strategies” foi adicionado ao código para aplicação deste padrão de projeto. Dentro deste diretório, existem 3 interfaces que definem uma assinatura que todas as estratégias correlatas devem implementar. As estratégias que implementam essas interfaces estão separadas no diretório “concrete” e dentro desse diretório as estratégias concretas são separadas em três outros diretórios “filter”, “sort” e “criterion”.



A primeira interface é a “SortStrategy”, que será implementada por duas estratégias diferentes que visam ordenar de forma crescente ou decrescente uma lista de produtos em Java, o InsertionSort e o QuickSort que serão utilizados no código para ordenar os produtos conforme estoque, preço ou descrição.

A segunda interface é a “FilterStrategy”, que é implementada pelas estratégias concretas que selecionam determinados produtos da lista segundo um filtro. As estratégias concretas são: “SameCategoryFilter”, que seleciona produtos de uma categoria passada via argumento no programa. “PriceBetweenFilter”, que seleciona produtos dentro de um intervalo de preços passado via uma String em argumento de programa que segue o formato “5.00,20.00” (nesse caso, os produtos selecionados estarão entre o 5.00 e o 20.00). “LessThanOrEqualStockFilter” que seleciona os produtos que tem uma quantidade no estoque menor ou igual ao valor passado no argumento do programa. “DescriptionContainsSubstringFilter” que irá selecionar os produtos que contêm uma string passada via argumento no programa. “EveryFilter” que seleciona todos os produtos.

A terceira interface é a “CriterionStrategy”, que será implementada por 6 classes que irão representar os possíveis critérios de ordenação do documento: ordem crescente e decrescente pela descrição do produto, ordem crescente e decrescente pela quantidade no estoque e ordem crescente e decrescente pelo preço do produto.

O padrão de projeto Decorator foi utilizado para as opções de formatação no programa. O diretório “decorator” contém o componente, o componente concreto, o decorator e o decorator concreto. O componente representa o contrato comum para todos os objetos que podem ser decorados, o componente concreto é a implementação do componente. O Decorator é a classe abstrata que implementa a interface do componente. A ideia é que o decorator adicione funcionalidades extras

sem uso de herança, apenas com o uso de composição, isso permite adicionar comportamentos extras sem modificar o código existente. Existem 3 classes Decorator: “BoldDecorator”, “ColorDecorator” e “FormattedDecorator” que estendem a classe abstrata “ProductDecorator”.

No diretório “main” estão as classes que vieram inicialmente no EP, as principais mudanças são a criação de uma classe “Main” separada da classe “GeradorDeRelatorios” visando respeitar o princípio da responsabilidade única das classes. A classe “MetodosAuxiliaresGerador” também foi adicionada para separar alguns métodos úteis que são utilizados pela classe “GeradorDeRelatorios”. Os produtos são carregados diretamente do arquivo csv em uma List do arcabouço de coleções de Java.