

Relatório – EP2 COO “Gerador de Relatórios”

Aluno: João Ricardo Paiva Camargo

Nº USP: 14747274

O relatório a seguir se refere a explicação e justificativa das mudanças feitas durante a refatoração do código de um gerador de relatórios, visando a aplicação dos padrões de código Strategy e Decorator, objetivo do segundo EP da disciplina “Computação Orientada a Objetos”.

Mudanças em “GeradorDeRelatórios”

As novas possibilidades de critérios de ordenação e filtragem foram adicionadas as variáveis de classe e tiveram as classes que as gerenciam criadas, como é requisitado pelo padrão Strategy.

Agora “produtos” não é mais armazenado como o vetor `Produto[]`, passando para um implementação de lista `List<ProdutoFormatacao> produtos`.

Os métodos “particiona” e “ordena” foram movidos para as classes na qual são utilizados, `InsertionSort` e `QuickSort`, removendo a necessidade de verificação do algoritmo escolhido, já que com o encapsulamento a chamada do método já resolve esse problema.

Foi criado o método `carregaProdutos`, permitindo que a lista de produtos seja lida de um arquivo `.csv`.

O método “geraRelatório” agora se beneficia das novas interfaces criadas, “Filtros”, “CritérioDeOrdenacao” e “MetodoDeOrdenacao”, tornando o código consideravelmente mais simples.

Além disso, outras mudanças básicas foram feitas para que o código se adeque as suas novas necessidades e seus padrões de código.

Padrão Strategy

O padrão Strategy foi aplicado de forma que agora os diferentes algoritmos de ordenação, critérios de ordenação e filtros estão divididos em interfaces que são implementadas pelas classes que a utilizam. Como exemplo, agora

“InsertionSort” e “QuickSort” implementam a interface “MetodoOrdenacao”. Isso acabar por facilitar a manutenção do código e tornar sua expansão mais simples.

Padrão Decorator

Como requisitado, agora a formatação do relatório não é mais responsabilidade da classe “GeradorDeRelatorios”, sendo agora gerenciado pelas novas classes que implementam “Produto” (“ProdutoCor”, “ProdutoItalico”, “ProdutoNegrito”). Agora também é possível imprimir um produto colorido graças a classe ProdutoCor.

Conclusão

Com todas as mudanças realizadas, agora espera-se que o Gerenciador de Relatórios seja um programa com um código mais moderno, completo, legível, de mais fácil manutenção e com uma maior possibilidade de expansão. Isso, claro, graças a aplicação dos padrões de código Strategy e Decorator e da utilização do arcabouço de coleções do Java. Essas mudanças são bem vindas em outros programas no qual esses padrões de código sejam aplicáveis, a fim de garantir um código mais limpo e seguro.