



Vamos praticar!

Vamos pensar em um sistema de validação de pagamentos, seja por cartão de crédito ou débito. Um cartão é composto por números na frente, um código de segurança e uma data de validade.

Por sua vez, o cartão pode ser de crédito ou de débito. Se for de crédito, terá um campo para o limite e outro para o saldo utilizado. Se for de débito, terá um único campo para o saldo disponível.

Para efetuar um pagamento com cartão, você precisa receber uma autorização. Se o cartão for de débito, a autorização ocorre se o saldo da conta em que o cartão é debitado for suficiente. No caso do cartão de crédito, ocorre se o limite não for ultrapassado.

Atualmente, temos dois serviços chamados: `ProcessadorCredito` e `ProcessadorDebito`. Ambos possuem um método para processar o pagamento, mas primeiro, validamos se a data de vencimento é posterior à data atual. Essa lógica é comum, independentemente de o pagamento ser feito a débito ou a crédito, ou seja, temos um código que se repete nos dois métodos. Os métodos que utilizam os dois processadores de pagamento, recebem o cartão e o valor a ser cobrado.

Gostaríamos de remover o código duplicado. Como você resolveria isso aplicando o Template Method pattern? Também gostaríamos de ver por console se o pagamento pode ser autorizado. Sendo assim, faça o diagrama UML e a implementação em JAVA.