3) Apenas pare efeito de entender melhor o abstract, remova (delete) o método atualiza() da ContaPoupanca, dessa

forma ele herdará o método diretamente de Conta.

Qual é o problema com a classe ContaPoupanca?

**RESPOSTA:Pelo que pude ver o problema com a classe ContaPoupanca ao remover o método atualiza é que ela passará a herdar o método diretamente da classe Conta, desde que a classe Conta não tenha declarado o método atualiza como abstrato. Então, se a classe Conta tiver o método atualiza() como abstrato e não fornecer uma implementação concreta para ele, isso gerará um erro de compilação na classe ContaPoupanca.**

5) Existe outra maneira de a classe ContaCorrente compilar se você não reescrever o método abstrato?

**RESPOSTA:Uma outra maneira que encontrei de a classe ContaCorrente compilar sem reescrever o método abstrato atualiza da classe Conta seria tornar a própria classe ContaCorrente abstrata e transferir a responsabilidade de implementação do método atualiza para suas subclasses concretas.**

6) Pra que ter o método atualiza na classe Conta se ele não faz nada? O que acontece se simplesmente

apagarmos esse método da classe Conta e deixarmos o método atualiza nas filhas?

**RESPOSTA:Ter o método atualiza na classe Conta mesmo que ele não faça nada pode ser útil para fornecer um método padrão que pode ser chamado em todas as subclasses, mas sem uma implementação padrão na classe pai. No entanto, se ele não tiver uma implementação significativa na classe Conta e todas as subclasses precisarem de implementações diferentes, pode ser mais limpo e claro remover o método atualiza() da classe Conta e implementá-lo diretamente nas subclasses.**

7) Posso chamar um método abstrato de dentro de um outro método da própria classe abstrata? Um exemplo: o

mostra do Funcionario pode invocar this.getBonificacao()?

**RESPOSTA:Sim, é possível chamar um método abstrato de dentro de outro método da própria classe abstrata.**