



Paulo Sérgio Medeiros
Técnicas de Programação 2

Exercício 1

Implemente as classes “Biblioteca”, “Livro”, “Cliente” e “Aluguel”. Um livro possui título, autor, editora e data de publicação, um cliente possui nome e CPF, e um Aluguel representa o evento de um Cliente alugando um Livro. Uma biblioteca possui livros em sua coleção, que pode ser de até 1000 livros, e a mesma permite que seus clientes os aluguem, de acordo com as seguintes regras:

- Um cliente só pode alugar 2 livros por vez;
- Um cliente só pode alugar um livro caso ele não seja um dos 3 últimos livros que ele alugou;
- A biblioteca só possui 1 exemplar de cada livro em sua coleção



Exercício 1

Além da lista de livros em sua coleção, uma biblioteca também deve manter uma lista atualizada de seus clientes.

Implemente os métodos cadastrar/remover livro, cadastrar/remover cliente, alugar livro, devolver livro, e outros que achar necessário para que representar corretamente o cenário descrito, e sinta-se a vontade pra criar os atributos que você achar necessário.

A sua biblioteca deve ser também capaz de emitir um relatório dos aluguéis realizados na biblioteca. Para isto deve receber um filtro por data (min. e max.) e livro ou cliente.

No final, tente identificar os relacionamentos existentes no cenário descrito, e classifique-os com relação a sua cardinalidade.

Exercício 2

Pense em 3 comportamentos que podem ser representados através de 3 interfaces diferentes. Em seguida, pense em objetos que compartilhem alguns desses comportamentos entre si, e crie classes para representá-los.

Crie um programa que instancie estes objetos e exerça os comportamentos definidos nas interfaces por meio de polimorfismo. Por fim, transforme uma das interfaces em classe abstrata. Justifique se a mudança, no conjunto de objetos e interfaces que você escolheu, faz sentido. Isto é, o conceito/entidade do mundo real seria melhor representado por meio de uma interface ou classe abstrata?

Exercício 2

exemplo, você deve criar a sua estrutura de classes e interfaces

