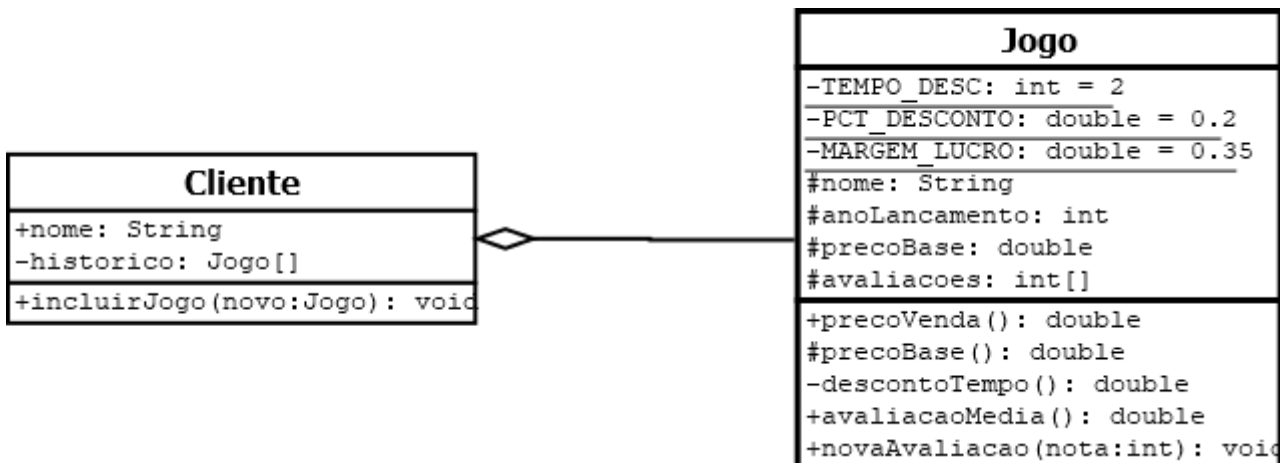


Instruções:

- Esta prova tem duas etapas: a primeira aqui no laboratório, a segunda para ser terminada posteriormente.
- As atividades atribuídas a cada etapa só serão corrigidas naquela etapa (ou seja, se você fizer uma questão da primeira etapa na segunda, estes pontos não contarão na nota da primeira etapa).
- Para a segunda etapa, você deve obrigatoriamente utilizar o código entregue na primeira etapa. Possíveis alterações e melhorias só contarão pontos na entrega da segunda etapa e precisam manter pelo menos a estrutura principal do código anterior.
- Entregue como solução, em cada etapa, um arquivo no formato zip contendo todos os arquivos .java utilizados. Não serão aceitos outros formatos, então certifique-se de que é um arquivo zip no envio.
- A segunda etapa estará disponível no Canvas a partir das 13h00 desta sexta, 01/04 e ficará disponível até às 23h59 do dia 07/04. Ao iniciar a segunda etapa, você tem até duas horas para concluir a entrega.
- Se você estiver fazendo prova com computador próprio, utilize o modo avião. Só é permitido manter abertas as janelas deste PDF e da IDE de programação.

1) Uma empresa irá iniciar seu negócio de venda de jogos digitais. Inicialmente, a empresa vende apenas jogos em modalidade de *download*. Um jogo tem um preço base para a sua venda, recebido da sua fabricante. A loja vende um jogo com uma margem de lucro de 35% sobre este preço. No entanto, a cada 2 anos após o lançamento, o preço base do jogo recebe um desconto de 20% do seu valor-base de venda. A loja, então, calculará a margem sobre este valor e assim fará a venda. Um jogo pode receber avaliações em notas que vão de 1 a 5 e deve saber responder sua nota média de avaliação. Os clientes compram jogos e isto fica armazenado em seu histórico de compras.

O modelo abaixo foi desenvolvido pelos analistas de sua companhia. Ele **não inclui** construtores ou métodos get/set, os quais você pode incluir se achar necessário, desde que get/set sigam o padrão *açãoAtributo()*, por exemplo, *getAnoLancamento()*. Se precisar criar atributos privados, isto também é permitido, mas não parece ser necessário.



PARTE 1: 01/04, 10h40 – 12h20, em laboratório

Implemente as classes Jogo e Cliente conforme especificados no diagrama, seguindo os requisitos necessários. No método *main* de uma classe App ou equivalente, crie um Cliente, um Jogo de 2022 e registre a venda do Jogo para este cliente. Altere o ano de lançamento do Jogo para 2019 e mostre seu novo valor de venda. **(12 pontos)**