UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE - UFRN

Instituto Metrópole Digital

IMD0040 - Linguagem de Programação 2

Atividade

Contextualização: Você foi chamado para desenvolver classes em Java para controle e gestão de um almoxarifado. O objetivo é o cadastro e controle de usuários, produtos, estoque e solicitações. Para concluir seu trabalho, você deverá realizar as tarefas abaixo relacionadas:

*IMPORTANTE: Informações de inserções e Mensagens (Informações, alertas e avisos) devem ser feitas utilizando a classe JOptionPane do pacote javax.swing, consulte a documentação.

Tarefa 01: Criar as seguintes Classes:

- Usuário:
  - Campos: nome e cpf
  - Todos os campos privados
  - Métodos de acesso e modificadores para todos os campos
  - Construtor com todos os campos como parâmetros
  - Construtor Vazio
- Produto: (0,5)
  - Campos: codigoProduto, nomeProduto , quantidade, preço e Hora.
  - Todos os campos privados
  - Métodos de acesso e modificadores para todos os campos, exceto modificador para o campo codigoProduto e hora, que deverá ser passado no construtor e ser iniciado por padrão com: PROD-<valorParâmetreo>-NumAleatorio (até 999)
  - Construtor com todos os campos como parâmetros

Observação: A quantidade está sendo passada na classe produto apenas por fins didáticos, não sendo recomendado esse procedimento.

- Estoque:
  - Campos: Lista de Produtos e Lista de Solicitações
  - Métodos:
    - Adicionar produtos no estoque (addProduto(Produto))
    - Remover produtos do estoque (removerProduto(Produto))
    - Listar todos os produtos do estoque (listarProdutos())
- SolictacaoEstoque:
  - Campos: Solicitante(Usuário), Produto, Quantidade e Data da Solicitação
  - Construtor com todos os campos como parâmetros
  - Criar apenas métodos de acesso: retornando o nome do usuário e o nome do produto.

Tarefa 02: Modificando comportamentos das classes:

- Estoque:
    - Método *addProduto*: Só é possível adicionar um novo produto no estoque, se a quantidade for maior que 0 (zero), caso contrário o sistema emitirá a seguinte mensagem: *Produto <nomedoproduto> com quantidade inválida.*
    - Método *removerProduto:* Só é possível remover um produto caso ele exista no estoque.
    - Método *solicitarProduto*: Para solicitar um produto no estoque, deve ser informado o usuário (objeto), produto (objeto) e a Quantidade, só é possível solicitar o produto caso exista e se sua quantidade for maior que zero. Caso contrário o sistema devera emitir uma mensagem: Produto <nomedoproduto> não existe no estoque ou não tem estoque disponível..
        - Caso a solicitação seja atendida, a quantidade do produto deve ser atualizada;
        - Caso a solicitação seja atendida, os dados da solicitação (usuário e produto) devem ser armazenados na lista de solicitações.
    - Método *listarSolicitacoes*: Listar todas as solicitações do estoque e a quantidade total de solicitações realizadas.
    - Método *buscarProdutos*: Buscar informações de um produto pelo nome, caso encontre o sistema mostrará a seguinte mensagem: Informações do Produto: <código do produto> <nomedo produto> <quantidade> e <Preço> , caso não exista o sistema emitirá a seguinte mensagem: Produto <nomedoproduto> não encontrado.
        - Este método deve ser construindo utilizando Iterator
        - As mensagens só devem aparecer uma única vez
        - Preço no formato da moeda brasileira.

<u>Orientações de Entrega da Avaliação:</u>

- Apenas os arquivos classes Java: Produto.java, Usuario.java, Estoque.java e SolictacaoEstoque.java;
- Comprimir os arquivos com formato ZIP;