| Uma imagem contendo placar, desenho, relógio  Descrição gerada automaticamente | **ATIVIDADE ADICIONAL PRÁTICA – WEB-JAVA** |
| --- | --- |
| web-java-4 |

**Instruções gerais:**

| 1. Utilize o Eclipse ou o STS para desenvolver os algoritmos. |
| --- |
| **Mantenha as entregas das Atividades em dia na Plataforma Canvas** |

**EXERCÍCIOS**

**Boas práticas:**

1. Leia o enunciado do exercício com atenção
2. Observe as indicações de Entrada e Saída esperadas em cada exercício
3. Observe com atenção os desenhos e diagramas inseridos nos exercícios para facilitar a compreensão
4. Utilize o Cookbook, os Vídeos da Plataforma e os Códigos guia como referências para a resolução dos exercícios
5. Caso ainda fique alguma dúvida, consulte os instrutores da sua turma pelo Discord

**Lista de exercícios**

**Exercícios sobre Springboot com relacionamento**

1. **Gerenciamento de Tarefas e Categorias**Desenvolva um projeto CRUD para gerenciar tarefas e suas categorias. As entidades "Tarefa" e "Categoria" devem ter atributos como título e descrição para tarefas, e nome e descrição para categorias, com um relacionamento de muitos-para-um entre tarefas e categorias.
2. **Controle de Inventário e Fornecedores**Crie um sistema CRUD para gerenciar produtos e seus fornecedores. As entidades "Produto" e "Fornecedor" devem incluir atributos como nome, descrição e preço para produtos, e nome, contato e endereço para fornecedores, com um relacionamento de muitos-para-um entre produtos e fornecedores.
3. **Cadastro de Clientes e Pedidos**Implemente um sistema CRUD para gerenciar clientes e seus pedidos. As entidades "Cliente" e "Pedido" devem ter atributos como nome e endereço para clientes, e data, valor total e status para pedidos, com um relacionamento de um-para-muitos entre clientes e pedidos.
4. **Gestão de Funcionários e Departamentos**Desenvolva um projeto CRUD para gerenciar funcionários e departamentos. As entidades "Funcionário" e "Departamento" devem incluir atributos como nome e cargo para funcionários, e nome e localização para departamentos, com um relacionamento de muitos-para-um entre funcionários e departamentos.
5. **Registro de Livros e Autores**Crie um sistema CRUD para gerenciar livros e autores. As entidades "Livro" e "Autor" devem ter atributos como título, ano de publicação e gênero para livros, e nome e biografia para autores, com um relacionamento de muitos-para-muitos entre livros e autores.
6. **Sistema de Reservas de Hotel e Quartos**Implemente um projeto CRUD para gerenciar reservas e quartos de hotel. As entidades "Reserva" e "Quarto" devem incluir atributos como nome do hóspede, data de entrada e saída para reservas, e número, tipo e preço para quartos, com um relacionamento de muitos-para-um entre reservas e quartos.
7. **Cadastro de Eventos e Participantes**Desenvolva um sistema CRUD para gerenciar eventos e participantes. As entidades "Evento" e "Participante" devem ter atributos como nome e data para eventos, e nome, e-mail e telefone para participantes, com um relacionamento de muitos-para-muitos entre eventos e participantes.
8. **Gestão de Cursos e Estudantes**Crie um projeto CRUD para gerenciar cursos e estudantes. As entidades "Curso" e "Estudante" devem ter atributos como título e descrição para cursos, e nome, matrícula e e-mail para estudantes, com um relacionamento de muitos-para-muitos entre cursos e estudantes.
9. **Controle de Despesas e Categorias de Despesas**Proponha um sistema CRUD para gerenciar despesas e suas categorias. As entidades "Despesa" e "CategoriaDespesa" devem incluir atributos como valor, data e descrição para despesas, e nome e descrição para categorias de despesas, com um relacionamento de muitos-para-um entre despesas e categorias.
10. **Cadastro de Produtos e Avaliações**Desenvolva um projeto CRUD para gerenciar produtos e suas avaliações. As entidades "Produto" e "Avaliação" devem ter atributos como nome, descrição e preço para produtos, e nota, comentário e data para avaliações, com um relacionamento de um-para-muitos entre produtos e avaliações.