| Uma imagem contendo placar, desenho, relógio  Descrição gerada automaticamente | **ATIVIDADE ADICIONAL PRÁTICA – JAVA** |
| --- | --- |
| JAVA-14 – Optional |

**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 Optional em Java**

1. **Uso Básico de Optional**:
   * Crie um método chamado buscarNome que receba um nome como parâmetro e retorne um Optional<String>. Se o nome for "admin", retorne o Optional contendo o nome. Caso contrário, retorne Optional.empty(). No método main, use o método buscarNome e, se o nome estiver presente, exiba-o.
2. **Tratamento de Valores Nulos com Optional**:
   * Crie uma classe Pessoa com atributos nome e email. Implemente um método chamado obterEmail que retorne um Optional<String>. No método main, crie uma Pessoa sem definir o email, utilize o Optional para verificar se o email está presente, e exiba uma mensagem adequada caso não esteja.
3. **Uso de orElse com Optional**:
   * Crie um método chamado obterEndereco que retorne um Optional<String> contendo o endereço de um usuário. Se o endereço for nulo, retorne um Optional.empty(). No método main, use orElse para fornecer um endereço padrão caso o valor não esteja presente e exiba o resultado.
4. **Uso de ifPresent com Optional**:
   * Crie uma classe Carro com um atributo modelo. Implemente um método buscarCarro que retorne um Optional<Carro>. No método main, utilize ifPresent para exibir o modelo do carro, caso ele esteja presente.
5. **Encadeamento de Métodos com Optional**:
   * Crie uma classe Livro com atributos titulo e autor. Crie um método buscarAutor que retorne um Optional<String> contendo o nome do autor. No método main, simule a busca de um autor e use métodos como map, filter, e flatMap para manipular e exibir o resultado, considerando que o autor pode não estar presente.