

# RESUMO DA AULA

## Desenvolvimento Gradual e Refatoração

---

### Desenvolvimento Gradual e Refatoração

O desenvolvimento gradual e a refatoração são duas práticas essenciais para manter o código limpo ao longo do tempo. O desenvolvimento gradual sugere que você não precisa tentar resolver tudo de uma vez. Em vez disso, você deve criar soluções simples que podem ser melhoradas ao longo do tempo.

A refatoração é o processo de reescrever o código sem modificar sua funcionalidade, mas melhorando sua estrutura e legibilidade. Refatorar periodicamente é importante para evitar que o código se torne confuso ou ineficiente.

Dicas para refatoração eficiente:

- 1 - Pequenas mudanças: Não tente refatorar um grande bloco de código de uma só vez. Refatore gradualmente, garantindo que cada mudança seja testada.
- 2 - Priorize simplicidade: Sempre que possível, busque simplificar o código. Remova duplicações, reduza funções longas e elimine variáveis desnecessárias.
- 3 - Teste sempre: Após refatorar, certifique-se de que o comportamento do código permaneça o mesmo, verificando com testes unitários ou funcionais.

Exemplo de refatoração:

Antes:

```
function calcularDesconto(preco, desconto) {  
  
  if (desconto > 0) {  
  
    return preco - (preco * desconto);  
  }  
}
```

```
} else {  
    return preco;  
}  
}
```

Após refatoração:

```
function calcularDesconto(preco, desconto) {  
    return desconto > 0 ? preco - (preco * desconto) : preco;  
}
```

---