

PROFESSOR JEFFERSON CHAVES

jefferson.chaves@ifpr.edu.br

ATIVIDADE AVALIATIVA

Funções

Funções são blocos de código **reutilizáveis** que realizam uma tarefa específica. Elas ajudam a modularizar um programa, tornando-o mais organizado, legível e fácil de testar e de manter. **É inimaginável um código maduro, sem o uso de funções.**

O que é escopo de funções?

O escopo define onde uma variável ou função pode ser acessada dentro do código. Em relação às funções, existem dois principais tipos de escopo:

- **Escopo Local:** Variáveis declaradas dentro de funções só podem ser acessadas dentro dessa função.
- **Escopo Global:** Variáveis declaradas fora de funções podem ser acessadas de qualquer lugar do código.

```
<?php

$contador = 0; // Variável global

function incrementar() {
    global $contador; // Acessa a variável global
    $contador++;
}

incrementar();
incrementar();

echo "Valor do contador: $contador"; // Saída: Valor do contador: 2
```

Exemplo de variável global com PHP

Por que se deve evitar variáveis globais?

Variáveis globais devem ser evitadas porque:

- Podem ser modificadas acidentalmente – Qualquer parte do código pode alterá-las, tornando o comportamento do programa imprevisível.
- Dificultam a manutenção – Como estão acessíveis em qualquer parte do código, fica difícil saber onde e por que foram alteradas.
- Atrapalham a reutilização do código – Funções que dependem de variáveis globais não podem ser facilmente reaproveitadas em outros projetos.
- Podem causar conflitos de nomes – Se várias partes do código usarem variáveis globais com o mesmo nome, podem ocorrer bugs difíceis de encontrar.
- **Melhor prática:** Use parâmetros e retorno de função em vez de variáveis globais!

O que é passagem de parâmetros por valor e por referência?

Os parâmetros podem ser passados para funções de duas maneiras:

- **Por valor** – Quando passamos um **tipo primitivo** (número, string, booleano, etc.), a função recebe uma **cópia** do valor, e alterações feitas dentro da função **não afetam a variável original**.
- **Por referência** – Quando passamos um **objeto, array ou outra estrutura complexa**, a função recebe um **ponteiro para o valor original**, permitindo que as alterações feitas dentro da função **afetem a variável original**.
 - **Cuidado!** O uso excessivo de passagem por referência pode tornar o código difícil de entender, pois a função pode modificar variáveis externas de forma inesperada.

Atividade

Usando como o base o algoritmo desenvolvido em sala, faça as seguintes modificações do programa:

1. Programe uma função menu, que permita ao usuário, criar se cadastrar, criar sua conta, depositar, sacar e consultar seu saldo.
2. Refatore a função de “cadastro de cliente”, garantindo que o CPF seja válido.
3. Refatore a função “depósito” para garantir que não sejam depositados valores negativos.
4. Refatore a função “saque”, para impedir saques maiores que o saldo em conta.