

SW-I

SISTEMAS WEB I

Prof. Anderson Vanin

Exercícios

1. Declare três variáveis que conterão respectivamente: nome de uma pessoa, idade e um valor (pode ser de salário ou de mesada), serão valores atribuídos, não digitados. Verifique o que foi impresso na tela abaixo e reproduza isto em um programa utilizando os dados das variáveis criadas. Os itens que estão sublinhados são valores de variáveis.

```
Olá! Qual o seu nome?  
Tyrone  
Oi Tyrone! Quantos anos você tem?  
18  
Então você tem 18!!!  
Quanto você ganha Tyrone?  
100  
100! Eu espero que seja por hora e não por ano!!!! rrsrrs....
```

Exercícios

2. Faça um programa PHP que defina uma constante PI com valor 3.141592. Crie uma variável, atribua um valor que represente o raio de uma esfera. Calcule e apresente sua área e seu volume, dados pelas fórmulas:

$$AREA = 4\pi R^2$$

$$VOLUME = \frac{4}{3}\pi R^3$$

Exercícios

3. Faça um programa PHP que crie duas variáveis e atribua dois valores inteiros. Apresente todas as operações: adição, subtração, divisão, multiplicação e resto da divisão utilizando os operadores de atribuição.

Formulários HTML com GET e POST em PHP

Introdução

Mostrar como capturar e processar dados enviados por formulários HTML utilizando os métodos **GET** e **POST** em PHP, explicando as diferenças entre eles e quando cada um deve ser utilizado.

Formulários HTML

Os formulários HTML permitem que os usuários enviem dados para o servidor. A tag **<form>** é utilizada para definir um formulário e os campos de entrada são definidos por meio de **<input>**, **<textarea>**, **<select>**, entre outros.

```
<form action="processar.php" method="POST">  
  <label for="nome">Nome:</label>  
  <input type="text" id="nome" name="nome">  
  <button type="submit">Enviar</button>  
</form>
```

Método GET

- Os dados são enviados pela URL.
- Limite de tamanho de dados (dependendo do navegador e servidor).
- Pode ser armazenado em cache e nos históricos do navegador.
- Não é recomendado para enviar informações sensíveis.

Método POST

- Os dados são enviados no corpo da requisição HTTP.
- Permite enviar grandes quantidades de dados.
- É mais seguro para informações sensíveis (embora não criptografado por padrão).
- Não fica armazenado na URL.

EXEMPLO PRÁTICO

Diferenças entre GET e POST

Característica	GET	POST
Visibilidade dos dados	Na URL	Oculto no corpo da requisição
Limite de dados	Pequeno	Grande
Armazenamento em cache	Sim	Não
Segurança	Baixa	Moderada
Uso recomendado	Parâmetros de busca, links compartilháveis	Formulários de login, envio de dados sigilosos

EXERCÍCIO PRÁTICO

Crie um formulário que permita ao usuário enviar um **nome** e uma **mensagem**. Os dados serão enviados via **POST** e serão exibidos em uma página separada.