

Explicação do HTML:

1. `<!DOCTYPE html>`

- Função: Declara o tipo de documento, informando ao navegador que este é um arquivo HTML5. Isso ajuda a garantir que o conteúdo seja interpretado corretamente.

2. `<html lang="pt-BR">`

- Função: Define o início do documento HTML e especifica que o idioma do conteúdo é o português do Brasil (pt-BR). Isso é importante para acessibilidade e para que os motores de busca interpretem corretamente o conteúdo.

3. `<head>`

- Função: A seção `<head>` contém metadados sobre o documento, como a codificação de caracteres, configurações de visualização e links para arquivos externos (CSS e outros).

4. `<meta charset="UTF-8">`

- Função: Define a codificação de caracteres como UTF-8, que é um formato universal que suporta todos os caracteres especiais (acentos, símbolos, etc.), essencial para o português e outros idiomas.

5. `<meta name="viewport" content="width=device-width, initial-scale=1.0">`

- Função: Faz com que a página seja responsiva. Configura o layout da página para ser adaptável a diferentes

tamanhos de tela, ajustando a largura para a largura do dispositivo e a escala inicial para 1 (sem zoom inicial).

6. <title>Sorteio de Nomes</title>

- Função: Define o título da página, que é exibido na aba do navegador. O título aqui é "Sorteio de Nomes".

7. <link rel="stylesheet" href="from.css">

- Função: Linka o arquivo externo de estilo CSS chamado from.css, que vai ser usado para estilizar o conteúdo da página. Este arquivo define o design e layout da página.

8. <link rel="stylesheet" href="hell.php">

- Função: Este trecho parece estar incorreto. O href deveria apontar para um arquivo CSS, mas está apontando para um arquivo PHP (hell.php). Um arquivo PHP não pode ser usado diretamente como folha de estilo. Se o arquivo for realmente um arquivo PHP, ele provavelmente não deveria ser referenciado dessa forma para estilizar a página. Deveria ser um arquivo CSS, como style.css.

9. <body>

- Função: A seção <body> contém o conteúdo visível da página web. Tudo que o usuário verá está dentro dessa tag.

10. <div class="container">

- Função: Cria uma "divisão" ou container para agrupar o conteúdo da página. A classe container pode ser usada para aplicar estilos (via CSS), como margem, largura e centralização do conteúdo.

11. <h1>Sorteio de Nomes</h1>

- Função: Exibe um título de nível 1 na página, que é o texto "Sorteio de Nomes". Isso geralmente é utilizado para destacar o título principal da página.

12. <form action="hell.php" method="POST">

- Função: Cria um formulário. O atributo action="hell.php" indica que, quando o formulário for enviado, os dados serão enviados para o arquivo PHP hell.php. O atributo method="POST" especifica que os dados serão enviados via método POST, o que significa que eles não serão visíveis na URL.

13. <label for="nomes">Digite os nomes separados por vírgula:</label>

- Função: Define um rótulo para o campo de entrada. O atributo for="nomes" associa este rótulo ao campo de entrada com o id nomes, ajudando na acessibilidade.

14. <textarea id="nomes" name="nomes" rows="5" placeholder="Exemplo: João, Maria, Ana"></textarea>

- Função: Cria uma área de texto onde o usuário pode digitar os nomes separados por vírgula. O atributo rows="5" define a altura do campo, com 5 linhas visíveis. O placeholder fornece um texto de exemplo dentro do campo para orientar o usuário.

15.

- Função: Cria uma quebra de linha para separar visualmente os elementos da página.

16. `<button type="submit">Sortear</button>`

- Função: Cria um botão que, quando clicado, envia o formulário. O `type="submit"` indica que esse botão envia os dados do formulário.

17. `</body>`

- Função: Marca o fim do conteúdo visível da página.

18. `</html>`

- Função: Marca o fim do documento HTML.

Explicação do CSS:

```
1. bodybody {  
  font-family: Arial, sans-serif;  
  margin: 0;  
  padding: 0;  
  background-color: #eef2f3;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  height: 100vh;  
  line-height: 1.6;  
  background-image: url(sus.png);  
  background-repeat: no-repeat  
  background-size: cover;}
```

- **font-family**: Define a fonte do texto como Arial, ou a fonte padrão sans-serif, caso Arial não esteja disponível.
- **margin e padding**: Remove as margens e o preenchimento padrão do body para evitar espaçamentos extras.

- **background-color:** Define a cor de fundo do corpo da página como um tom claro (#eef2f3).
- **display: flex:** Usa o modelo de layout Flexbox para alinhar os itens dentro do body.
- **justify-content: center:** Centraliza o conteúdo horizontalmente.
- **align-items: center:** Centraliza o conteúdo verticalmente.
- **height: 100vh:** Define a altura do body para 100% da altura da janela de visualização (viewport).
- **line-height:** Define a altura da linha como 1.6, o que melhora a legibilidade do texto.
- **background-image:** Define uma imagem de fundo (sus.png), além da cor de fundo.
- **background-repeat:** Impede que a imagem de fundo se repita.
- **background-size: cover:** Faz a imagem de fundo cobrir toda a área disponível do body, sem distorção.

2. .container

```
.container {
  background: rgba(31, 1, 102, 0.596);
  padding: 20px;
  border-radius: 8px;
  box-shadow: 0 5px 10px rgba(0, 0, 0, 0.1);
  text-align: center;
  width: 100%;
  max-width: 400px;
}
```

- **background:** Define um fundo com cor semi-transparente (azul escuro com opacidade 0.596).
- **padding:** Adiciona um preenchimento de 20px em todas as direções dentro do container.
- **border-radius:** Arredonda os cantos do container em 8px.
- **box-shadow:** Adiciona uma sombra sutil ao redor do container para dar um efeito de elevação (sombra suave de 5px no eixo y e 10px de difusão).
- **text-align:** Alinha o texto dentro do container ao centro.

- **width:** Define a largura do container como 100% da largura disponível.
- **max-width:** Define a largura máxima do container como 400px, para que ele não ultrapasse esse valor em telas maiores.

3. h1

```
h1 {  
  margin-bottom: 15px;  
  color: #ffffff;  
}
```

- **margin-bottom:** Adiciona um espaçamento de 15px abaixo do título <h1>.
- **color:** Define a cor do texto como branco (#ffffff).

4. label

```
label {  
  display: block;  
  margin-bottom: 10px;  
  color: #ffffff;  
}
```

- **display: block:** Faz com que o rótulo (label) seja exibido como um bloco, ocupando toda a largura disponível.
- **margin-bottom:** Adiciona um espaçamento de 10px abaixo do rótulo.
- **color:** Define a cor do texto do rótulo como branco (#ffffff).

5. textarea

```
textarea {  
  width: 95%;  
  padding: 10px;  
  margin-bottom: 15px;  
  border: 1px solid #ccc;  
  border-radius: 8px;
```

```
font-size: 14px;  
resize: none;  
}
```

- **width**: Define a largura da área de texto como 95% da largura disponível do elemento pai.
- **padding**: Adiciona 10px de preenchimento dentro da área de texto.
- **margin-bottom**: Adiciona um espaçamento de 15px abaixo da área de texto.
- **border**: Define uma borda fina de 1px e cor cinza clara (#ccc).
- **border-radius**: Arredonda os cantos da área de texto em 8px.
- **font-size**: Define o tamanho da fonte como 14px.
- **resize: none**: Impede que o usuário redimensione a área de texto.

6. button

```
button {  
  background-color: #4caf50;  
  color: white;  
  border: none;  
  padding: 10px 20px;  
  font-size: 16px;  
  border-radius: 4px;  
  cursor: pointer;  
}
```

- **background-color**: Define a cor de fundo do botão como verde (#4caf50).
- **color**: Define a cor do texto do botão como branco.
- **border**: Remove qualquer borda padrão do botão.
- **padding**: Adiciona 10px de preenchimento no eixo vertical e 20px no eixo horizontal.
- **font-size**: Define o tamanho da fonte do botão como 16px.
- **border-radius**: Arredonda os cantos do botão em 4px.
- **cursor**: Define o cursor como "pointer", indicando que o botão é clicável.

7. button: hover

```
button: hover {  
  background-color: #45a049;  
}
```

- **background-color:** Altera a cor de fundo do botão para um verde mais escuro (#45a049) quando o usuário passa o cursor sobre o botão (efeito de hover).

8. .result

```
.result {  
  margin-top: 20px;  
  font-size: 18px;  
  font-weight: bold;  
  color: #ffffff;  
}
```

- **margin-top:** Adiciona um espaçamento de 20px acima do elemento com a classe .result.
- **font-size:** Define o tamanho da fonte como 18px.
- **font-weight:** Define a fonte como negrito.
- **color:** Define a cor do texto como branco.

9. .error

```
.error {  
  color: red;  
  font-weight: bold;  
}
```

- **color:** Define a cor do texto como vermelha, geralmente usada para mensagens de erro.
- **font-weight:** Define a fonte como negrito, para dar mais destaque à mensagem de erro.

Explicação do PHP:

1. `if ($_SERVER['REQUEST_METHOD'] == 'POST')`

- **Função:** Verifica se o método de requisição HTTP é POST. Ou seja, o código só será executado se o formulário HTML tiver sido enviado com o método POST.

2. `$nomes = isset($_POST['nomes']) ? trim($_POST['nomes']) : '';`

- **Função:** Verifica se o campo nomes foi enviado via POST.
 - Se o campo nomes existir em `$_POST`, ele faz um `trim()` nos valores (removendo espaços extras no início e no final) e os armazena na variável `$nomes`.
 - Se o campo não existir, a variável `$nomes` recebe uma string vazia (`' '`).

3. `if (!empty($nomes))`

- **Função:** Verifica se a variável `$nomes` não está vazia. Ou seja, se o campo de entrada de nomes foi preenchido com algum valor.

4. `$listaNomes = array_map('trim', explode(',', $nomes));`

- **Função:**
 - A função `explode(',', $nomes)` divide a string de nomes (separados por vírgulas) em um array. Por exemplo, "João, Maria, Ana" se tornaria [`'João'`, `'Maria'`, `'Ana'`].
 - A função `array_map('trim', ...)` aplica a função `trim()` em cada item do array, removendo espaços extras em cada nome.

5. `if (count($listaNomes) > 1)`

- **Função:** Verifica se o número de nomes na lista é maior que 1. Ou seja, se há pelo menos dois nomes para sortear.

6. `$sorteado = $listaNomes[array_rand($listaNomes)];`

- **Função:**
 - A função `array_rand($listaNomes)` seleciona aleatoriamente um índice do array `$listaNomes`.
 - Esse índice é usado para pegar um nome aleatório do array e armazená-lo na variável `$sorteado`.

7. `echo "<p class='result'>Nome sorteado: $sorteado</p>";`

- **Função:** Exibe o nome sorteado dentro de um parágrafo com a classe `result`. O nome sorteado é mostrado em azul dentro de uma tag ``.
 - Isso gera uma resposta visível ao usuário, informando qual nome foi sorteado.

8. `else { echo "<p class='error'>Por favor, insira pelo menos dois nomes.</p>"; }`

- **Função:** Caso a lista de nomes tenha 1 ou menos elementos, exibe uma mensagem de erro solicitando que o usuário insira pelo menos dois nomes.

9. `else { echo "<p class='error'>Por favor, insira nomes válidos.</p>"; }`

- **Função:** Caso a variável `$nomes` esteja vazia (ou seja, o campo de entrada foi deixado em branco), exibe uma mensagem de erro pedindo para inserir nomes válidos.

```
10.else { echo "<p class='error'>Acesso inválido.  
Volte para a página inicial.</p>"; }
```

- **Função:** Se a requisição não foi feita via POST (por exemplo, se o formulário não foi enviado corretamente), exibe uma mensagem de erro informando que o acesso foi inválido e sugerindo voltar para a página inicial.

Código em Portugal:

Inicio

Caractere:nomes, listaNomes, sorteado;

Escrever("Digite os nomes para serem sorteados:");

Ler(nomes);

Se (MetodoRequisicao() = "POST") Então

nomes <- ValorDeEntrada("nomes");

Fim se:

Se (nomes != "") Então

listaNomes <- SepararTexto(nomes, ",");

Para i de 1 até Tamanho(listaNomes) faça

listaNomes[i] <- RemoverEspacos(listaNomes[i]);

FimPara:

Se (Tamanho(listaNomes) > 1) Então

sorteado <- listaNomes[Aleatorio(1, Tamanho(listaNomes))];

Escrever("Nome sorteado: ", sorteado);

Senão

Escrever("Por favor, insira pelo menos dois nomes.");

FimSe;

Senão

Escrever("Por favor, insira nomes válidos.");

FimSe;

Senão

Escrever("Acesso inválido. Volte para a página inicial.");

FimSe;

Fim.

Para acessar o site abra o terminal após abrir o arquivo do php e coloque o comando: php -S localhost:8080 -t.

Após o uso do comando copie a URL e coloque ela no navegador.

Feito por:Sandro, Dereck e Erick.