

[◀ VOLTAR](#)

Criação e tratamento de Cookies

Neste tópico apresentaremos como criar, manipular e resgatar dados através de cookies.

NESTE TÓPICO



Marcar
tópico



Cookies

Cookies são arquivos de texto armazenados no computador do cliente e são mantidos de propósito para seu rastreamento (DALL'OGGIO, P., 2009).

Tecnicamente falando, um cookie é uma pequena quantidade de informação, limitada, normalmente a 4KB e que existirá durante um período pré-definido (MUTO, C. A., 2006).

Há três etapas envolvidas na identificação de cookies:

- O servidor envia um arquivo com informações para o navegador. Por exemplo nome do cookie, valor, validade, caminho, domínio, segurança, etc.
- O navegador armazena essas informações na máquina local para uso futuro.
- Na próxima vez que o cliente acessar o site, com posse dessas informações o servidor consegue identificar facilmente o usuário.

O que podemos fazer com os cookies?

Com os cookies podemos fazer "tudo" o que quisermos, basta imaginar e programar, mas as principais utilizações de seu uso, são:

- Verificar se um usuário já logou no site (porém, neste caso, a melhor, mais e eficaz e segura maneira de se verificar isto, é utilizando o objeto session);
- Verificar se um usuário já votou numa enquete;
- Armazenar produtos de um carrinho de compras.

A anatomia de um cookie

Os cookies são geralmente fixados em um cabeçalho HTTP. Para entender como essa troca de informação é feita, vamos criar um cookie com o PHP, felizmente você só precisa aprender uma função para manipular cookies (MUTO, C. A., 2006)! Essa função é a **setcookie()** do PHP. Vejamos sua sintaxe:

```
1. <?php
2.     setcookie(name, value, expire, path, domain, secure);
3. ?>
```

Somente o parâmetro *name* é obrigatório, os demais parâmetros são opcionais.

Detalhando os argumentos temos:

- **name**: define o nome do cookie e é armazenado em uma variável de ambiente chamada HTTP_COOKIE_VARS. Esta variável é usada ao acessar cookies.
- **value**: define o valor da variável chamada e é o conteúdo que você realmente deseja armazenar.
- **expire**: - especifica a “tempo de vida” em segundos, após esse cookie tempo vai se tornar inacessível. Se esse parâmetro não estiver definido, então cookie expirará automaticamente quando o navegador for fechado.
- **path**: especifica os diretórios para o qual o cookie é válido. Um único caractere de barra permite que o cookie seja válido para todos os diretórios.
- **domain**: pode ser usado para especificar o nome de domínio em grandes domínios e deve conter, pelo menos, dois períodos válidos. Todos os cookies são válidos apenas para o anfitrião e de domínio que os criou.
- **secure**: pode ser definido como 1 para especificar que o cookie deve ser enviado por transmissão segura usando HTTPS de outro modo definido para 0, o que significa cookie pode ser enviado por HTTP regular.

Vamos a um exemplo simples de como criar e depois recuperar as informações de um *cookie*.

```
1. <?php
2.     // Cria um cookie chamado 'Usuario' com o valor 'Alguem'
3.     setcookie('Usuario', 'Alguem');
4.
5.     // Cria o mesmo cookie acima só que irá durar três dias
6.     setcookie('Login', 'acesso', (time() + (3 * 24 * 3600)));
7.
8.     // Cria o novo cookie para durar vinte minutos
9.     setcookie('carrinho', 'armazenar', (time() + (1200)));
10. >>
```

Se não definirmos o tempo de vida (expire) de um cookie, ele irá durar o tempo que o navegador estiver aberto. Para definir o tempo de vida basta somarmos o número de segundos desejados ao resultado da função `time()`, por exemplo: duas horas são 7200 segundos, então ficaria: `7200 + time()`, ou `(3600 * 2) + time()`. Lembrando que uma hora tem 3600 segundos.

Os cookies trabalham diretamente no cabeçalho do protocolo HTTP, ao acessarmos seu conteúdo teremos algo semelhante a isto, no exemplo abaixo, estamos utilizando como navegador o Google Chrome:

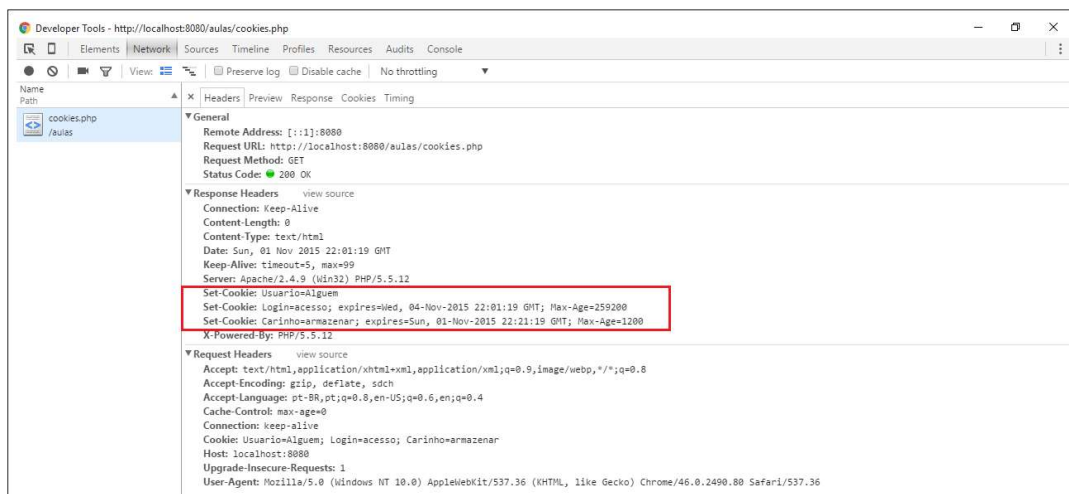


Figura 1: Exemplo de requisição HTTP utilizando o Google Chrome.

Fonte: Fonte: Prof. Adriano Milanez.

Vamos a outro exemplo de utilização de cookies:

```

1.  <?php
2.      if (($_SERVER["REQUEST_METHOD"] == "POST") || (isset($_COOKIE["voto"]))) {
3.          if (!isset($_COOKIE["voto"])) {
4.              setcookie('voto', 'true', (time()+(2*24*3600)));
5.              $voto = $_POST["musica"];
6.              echo "Você votou em: $voto <br>";
7.              echo "Obrigado pelo seu voto";
8.          }
9.          else {
10.             echo "Você não pode votar mais de uma vez!";
11.          }
12.      }
13.      else {
14.      ?>
15.          <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
16.              Vote no seu estilo de música preferido:<br>
17.              <br>
18.              <label>Forró</label><input type="radio" name="musica" value="forro" />
19.              <br />
20.              <label>Lambada</label><input type="radio" name="musica" value="lambada" /><br />
21.              <label>Pagode</label><input type="radio" name="musica" value="Pagode" /><br />
22.              <label>Rock</label><input type="radio" name="musica" value="rock" /><br />
23.              <label>Sertanejo</label><input type="radio" name="musica" value="sertanejo" /><br />
24.              <label>Samba</label><input type="radio" name="musica" value="samba" /><br />
25.              <input type="submit" value="Votar"/>
26.          </form>
27.      <?php } ?>

```

No código acima, vale chamar atenção para as linhas 2 e 3, principalmente quando usamos `$_COOKIE["voto"]`, essa instrução faz com que façamos a leitura do cookie "voto", atribuído na linha 4, com a instrução `setcookie`.

Na primeira que acessamos a página, teremos o seguinte resultado:



Figura 2: Exemplo de utilização de cookies.

Fonte: Fonte: Prof. Adriano Milanez.

Ao selecionar uma opção e clicar no botão "votar", teremos como resposta:

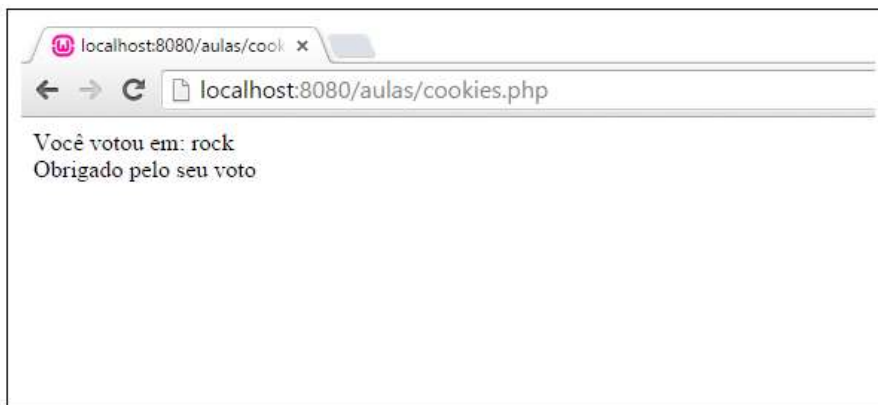


Figura 3: Exemplo de utilização de cookies.

Fonte: Fonte: Prof. Adriano Milanez.

Se tentarmos votar novamente, receberemos como resposta a tela abaixo:



Figura 4: Exemplo de utilização de cookies.

Fonte: Fonte: Prof. Adriano Milanez.

```
1. <?php
2.     setcookie('nome_cookie');
3. ?>
```

Se por algum motivo você precisar deletar algum cookie é só não definir um valor para ele, fazendo dessa forma:

Atenção: *Cookies* são parecidos com *Sessions* no fato de trabalharem com headers HTTP, então eles **precisam** ser definidos e manipulados antes de você enviar qualquer HTML para o navegador do visitante ou você receberá um erro (TANSLEY, D., 2006). Outro ponto importante é que, por esse fator, os cookies só estarão disponíveis para leitura (através da variável `$_COOKIE`) após o próximo carregamento de página.

Espero que o conteúdo tenha sido proveitoso. Estudem e até o próximo tópico!

Quiz

Exercício Final

Criação e tratamento de Cookies

INICIAR ➤

Referências

CONVERSE, T. & PARK, J. **PHP 5 - A Bíblia**. 2. Ed. Rio de Janeiro: Campus, 2005.

DALL'OGGIO, P. **Programando com orientação a objetos (Inclui *Design Patterns*)**. São Paulo: Novatec, 2009.

MARTIN, J. **Princípios de análise e projeto baseado em objetos**. Rio de Janeiro: Editora Campus, 1994.

MATHEUS, D. **Variáveis e constantes no PHP**. Disponível em <<http://www.diogomatheus.com.br/blog/php/variaveis-e-constantes-no-php/>>, acessado em 19/09/2015, às 12h15min. Publicado em 26/09/2012.

MUTO, C. A. **PHP e MySQL: Guia Introductório**. 3. Ed. Rio de Janeiro: Brasport, 2006.

NIEDERAUER, J. **Desenvolvendo Websites com PHP**. 2. Ed. São Paulo: Novatec, 2004.

PHP. **PHP: setcookie**. Disponível em <<http://br.php.net/setcookie>>. Acessado em 01/11/2015, às 21h05min.

SOARES, L.; AUGUSTO, B. **Aprendendo a Linguagem PHP**. 1. Ed. Rio de Janeiro: Ciência Moderna, 2007.

SOARES, W. **PHP 5 - Conceitos, Programação e Integração com Banco de Dados**. 4. Ed. São Paulo: Érica, 2004.

TANSLEY, D. **Como criar Web Pages rápidas e eficientes usando PHP e MySQL**. 1. Ed. Rio de Janeiro: Ciência Moderna, 2002.

W3Schols. **PHP Functions**. Disponível em: <http://www.w3schools.com/php/php_functions.asp>, acessado em 26/10/2015, as 21h25min.



Avalie este tópico



ANTERIOR

Diferenças entre Cookies e Sessões



Índice

Biblioteca
(https://www.uninove.br/conheca-a-uninove/biblioteca/sobre-a-biblioteca/apresentacao/) Portal Uninove
(http://www.uninove.br) Mapa do Site

Ajuda?
PRÓXIMO
(https://ava.uninove.br/ava/criacao-e-tratamento-de-sessoes/)

Criação e tratamento de Sessões

© Todos os direitos reservados

