

[◀ VOLTAR](#)

Diferenças entre Cookies e Sessões

Neste tópico apresentaremos as principais diferenças, vantagens e desvantagens de trabalhar com Cookies e Sessions.

NESTE TÓPICO



Marcar
tópico



Cookies e Sessions

Levando em consideração que o protocolo HTTP é *stateless*, ou seja, ele não mantém informações sobre a página navegada e estas serão apagadas quando cliente enviar a página ou fechar o navegador, toda nova interação que o seu cliente fizer acarretará em uma nova requisição e consequentemente uma nova resposta (CONVERSE, T. & PARK, J., 2005). Essas requisições são independentes e possuem um tempo de vida (conexão, envio de mensagem, resposta, encerramento da conexão). O servidor web não é capaz de identificar se duas requisições vieram de um mesmo navegador (SOARES, W., 2004).

É para suprir esta necessidade que entram os *cookies* e *sessions*.

A Netscape desenvolveu uma solução no seu navegador conhecida como *cookies* – tecnicamente falando, um cookie é uma pequena quantidade de informação, limitada, normalmente a 4KB e que existirá durante um período pré-definido (NIEDERAUER, J., 2004). Essas informações serão enviadas ao site de origem toda vez que uma nova página foi requerida. Cada cookie só pode ser lido pelo site que o tinha escrito, o que significa que é um modo seguro para armazenar informações entre as páginas.

Os *cookies* ganharam uma imagem ruim porque permitia saber quantas vezes um visitante veio ao seu site, ou o que eles fizeram no site, entre outras coisas; muitas pessoas acreditavam que os *cookies* sinalizavam o fim da privacidade na web, outros ainda acreditavam que os *cookies* podiam ler qualquer informação do seu disco rígido, e as pessoas foram encorajadas a desativar os *cookies* através da configuração do navegador. A realidade é que os *cookies* são relativamente inofensivos e agora são comumente aceitos.

As *sessions* nasceram da necessidade de se armazenar informações do cliente do lado servidor. Mas porque armazenar dados do cliente no servidor e não no cliente? Simples, porque os usuários são capazes de mexer onde eles quiserem e assim, poderiam modificar as informações armazenadas.

Já percebemos que tantos *cookies* quanto as *sessions* fazem a mesma coisa, armazenam informações dos clientes.

As informações arquivadas num cookie podem ficar armazenadas por longos períodos de tempo, variando de segundos até anos, após terem sido criadas (CONVERSE, T. & PARK, J., 2005).

As *sessions* são armazenadas no servidor, o que significa que os usuários não têm acesso à informação que você armazena sobre eles, isto é particularmente importante se você armazenar informações, como por exemplo, sobre o *login* do usuário, ou ainda, podemos armazenar informações dos produtos inseridos num carrinho de compras ou outras informações que você não quer que seu visitante seja capaz de editar manualmente. Nas *sessions* não existe limitação de tamanho como nos *cookies* uma vez que elas são armazenadas no servidor, porém a *session* está limitada ao tempo que o navegador estiver aberto ou ao seu *time-to-live*, tempo de vida, quando o navegador for encerrado ou o tempo de vida expirar, automaticamente a *session* é encerrada.

Cookies

Um cookie, como já mencionado, é uma pequena quantidade de informação armazenada no computador do cliente e que contém dados que serão utilizados em algum momento (NIEDERAUER, J., 2004). Alguns clientes ainda acreditam que por um motivo ou outro que os *cookies* são mal-intencionado, e acabam configurando o navegador para rejeitar *cookies*, e quanto a isso não há que possamos fazer. O único perigo que um cookie pode realmente representar é que eles podem armazenar informações sobre você sem você perceber, um *web site* pode, por exemplo, acompanhar quantas vezes você o acessou, ou quantas vezes você o visitou pelo *banner* que você clicou, etc. No entanto, eles não podem ler o número do seu cartão de crédito, invadir sua geladeira, ou qualquer coisa do tipo!

Ao criar *cookies*, devemos especificar seu tempo de vida, e, uma vez feito, o cookie permanece no local até essa data, quando "expira".

Uma vantagem dos *cookies* é que eles são descentralizados, ou seja, não precisamos nos preocupar com a criação de uma base de dados para armazenar informações. Como tal, os *cookies* são bons para pequenas

quantidades de informação para lidar, uma vez que isso envolve as preferências do usuário. Por exemplo, usamos *cookies* para armazenar informações de como os usuários querem a classificação dos seus *emails* ou ainda, as informações sobre qual time de futebol deverão ser exibidas, etc.

Agora vamos supor, que estejamos armazenando informações sobre o endereço de *email* de seu usuário, neste caso, é mais simples utilizarmos um banco de dados, pois este tipo de informação é algo que será armazenado por longo período de tempo, ao passo que os *cookies* são normalmente informações mais descartáveis, utilizadas por um curto espaço de tempo.

Um outro ponto importante a ser discutido, é que se você está armazenando informações pessoais em *cookies*, pense na segurança desse arquivo, como por exemplo: criptografia.

A criação e a manipulação de *cookies* será estudada no Tópico XII – Criação e tratamento de *cookies*.

Session

As *sessions* são uma combinação de um "arquivo" do lado do servidor e um "arquivo" do lado do cliente, com o arquivo do lado do cliente, contendo nada mais do que uma referência para os dados corretos no servidor (NIEDERAUER, J., 2004). Assim, quando o cliente visitar o *site*, o navegador envia o código de referência para o servidor, o qual carrega os dados correspondentes.

Nota: o "arquivo" aqui referenciado, não é arquivo como o temos nos *cookies* é apenas uma expressão que utilizamos para mostrar que teremos os dados armazenados em algum lugar, tanto no servidor quanto no cliente, portanto não existe, de fato, um arquivo com os dados de uma sessão para ser manipulados. A criação e a manipulação de *sessions* será estudada no Tópico XIII – Criação e tratamento de sessões.

Isto pode parecer um tanto quanto estranho, termos dois "arquivos", um "arquivo" do lado do cliente e outro do lado do servidor com todos os seus dados, mas há algumas vantagens:

- Uma *session* do lado do servidor pode conter quantidades enormes de dados sem nenhum incômodo para o cliente, lembrando que os *cookies*, do lado do cliente, são limitados em 4KB de tamanho.
- Seu *cookie* do lado do cliente contém apenas uma pequena porção de informação, como este arquivo é enviado ao servidor cada vez que alguém visita uma página em seu site, você está utilizando largura de banda desnecessariamente, pois poderíamos evitar tráfego na rede utilizando *sessions*.
- Os dados de uma *session* são muito mais seguros, somente o desenvolvedor é capaz de manipulá-lo, ao contrário dos *cookies* que são editáveis por todos.

Também é importante notar que as sessões duram apenas até que o usuário fechar seu navegador, enquanto que os cookies podem ser configurados para durar mais tempo (CONVERSE, T. & PARK, J., 2005).

Então, como você pode perceber, cada um tem suas próprias vantagens, mas possivelmente você terá de optar por um ou por outro, ou ainda, pela utilização dos dois: você quer que seus dados fiquem disponíveis no dia seguinte? Se sim, então sua única opção são os *cookies*. Claro que dependendo da informação a ser utilizada a melhor opção é armazená-la num banco de dados. Agora se você não precisa que sua informação esteja disponível sempre, as *sessions* são a melhor opção, isso sem contar que as *sessions* são mais fáceis de se trabalhar e não exigem que seus dados sejam enviados em conjunto com cada página, e também são apagadas assim que o visitante fechar seu navegador web, diferentemente dos *cookies*.

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

Quiz

Exercício Final

Diferenças entre Cookies e Sessões

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.

W3SCHOLS. **PHP: Sessions.** Disponível em: <http://www.w3schools.com/php/php_sessions.asp>, acessado em 31/10/2015, as 21h25min.



® Todos os direitos reservados