



# **Curso de Operador de Tecnologia da Informação e Comunicações**

# **Desenvolvimento Web**

## **PHP - Cookies**



Objetivo de  
Aprendizado

Entender como funcionam dos cookies e o comando include em páginas PHP.



A linguagem PHP é um das mais utilizadas no mundo devido a sua versatilidade e também por causa da facilidade de aprendizado. Existem várias ferramentas que podem ser utilizada com o PHP relacionadas a frameworks que podem ajudar no desenvolvimento rápido e seguro.

# Cookies em PHP



“Cookie é um grupo de dados trocados entre o navegador e o servidor de páginas, colocado num arquivo (ficheiro) de texto criado no computador do utilizador. A sua função principal é a de manter a persistência de sessões HTTP. (...) Um exemplo é aquele cookie que um site cria para que você não precise digitar sua senha novamente quando for ao site outra vez. Outros sites podem utilizá-los para guardar as preferências do usuário, por exemplo, quando o sítio lhe permite escolher uma cor de fundo para suas páginas”.

Como dito na definição wikipediana, cookies podem ser usados para armazenar dados do visitante, por exemplo: o idioma escolhido pelo visitante.

O PHP suporta de forma transparente **cookies HTTP**. Cookies são um mecanismo para armazenar dados no navegador remoto e assim rastrear ou identificar usuários que retornam. Você pode criar cookies usando as funções **setcookie()** ou **setrawcookie()**. Cookies fazem parte do cabeçalho HTTP, logo, **setcookie()** deve ser chamada antes que qualquer saída seja enviada ao navegador. Esta é a mesma limitação que a função **header()** tem. Você pode usar as funções de **buffering** de saída para atrasar as impressões no script até que você tenha decidido se vai ou não configurar qualquer cookie ou enviar quaisquer cabeçalhos.

Para criar um cookie:

```
<?php
//Setting new cookie
//formato do cookie: setcookie("name","value",time()+$int);

$value = 'alguma coisa de algum lugar';
setcookie("CookieTeste", $value);
setcookie("CookieTeste", $value, time()+3600); /* expira em 1 hora */
/*
name is your cookie's name
value is cookie's value
$int is time of cookie expires
*/
?>
```

Para recuperar um valor de cookie:

```
<?php
// Mostra um cookie individual
    echo $_COOKIE["your cookie name"];
// Outra maneira de depurar(debug)/testar é vendo todos os cookies
print_r($_COOKIE);
?>
```

Para fazer um cookie durar alguns dias:

```
<?php
    setcookie( "TestCookie", $value, strtotime( '+30 days' ) );

?>
```

Para deletar um cookie:

```
<?php
    unset($_COOKIE["yourcookie"]);

/*Or*/

    setcookie("yourcookie","yourvalue",time()-1);
    /*it expired so it's deleted*/

?>
```

Atualizando um cookie:

```
<?php
    setcookie("color","red");
    echo $_COOKIE["color"];
    /*color is red*/

    setcookie("color","blue");
    echo $_COOKIE["color"];
    /*new color is blue*/
?>
```

## “include” e “require” no PHP



A declaração include inclui e avalia o arquivo informado.

Os arquivos são incluídos baseando-se no caminho do arquivo informado ou, se não informado, o include\_path especificado. Se o arquivo não for encontrado no include\_path, a declaração include checará no diretório do script que o executa e no diretório de trabalho corrente, antes de falhar. O construtor include emitirá um aviso se não localizar o arquivo; possui um comportamento diferente do construtor require, que emitirá um erro fatal.

Se um caminho for definido — seja absoluto (iniciando com a letra do drive ou \ no Windows, ou / no Unix/Linux ), ou relativo ao diretório atual (começando com . ou ..) — o include\_path será completamente ignorado. Por exemplo, se o nome do arquivo iniciar com ../, o interpretador irá procurar pelo arquivo no diretório pai.

Para mais informações de como o PHP trabalha ao incluir arquivos e o caminho de inclusão, veja a documentação do include\_path.

Quando um arquivo é incluído, o código herda o escopo de variáveis da linha que a inclusão ocorrer. Qualquer variável disponível no arquivo que incluiu estará disponível no arquivo incluído, daquela linha em diante. Entretanto, todas as funções e classes definidas no arquivo incluído estarão no escopo global.

Exemplo:

```
//Arquivo vars.php

<?php

$color = 'green';
$fruit = 'apple';

?>
```

```
//Arquivo test.php

<?php

echo "A $color $fruit"; // os valores são mostrados?
include ("vars.php");

echo "A $color $fruit"; // A green apple

?>
```

### Comando “require”

A declaração require é idêntica a **include** exceto que em caso de falha também produzirá um erro fatal de nível E\_COMPILE\_ERROR. Em outras palavras, ele parará o script enquanto que o include apenas emitirá um alerta (E\_WARNING) permitindo que o script continue.

```
//Arquivo test.php

<?php

echo "A $color $fruit"; // os valores são mostrados?

require ("vars.php");

echo "A $color $fruit"; // A green apple

?>
```

### Comando “require\_once”

A declaração **require\_once** é idêntica a requirem exceto que o PHP verificará se o arquivo já foi incluído, e em caso afirmativo, não o incluirá (exigirá) novamente.

```
//Arquivo test.php

<?php

echo "A $color $fruit"; // os valores são mostrados?

require_once ("vars.php");

echo "A $color $fruit"; // A green apple

?>
```

### Referência

<http://blog.thiagobelem.net/trabalhando-com-cookies-no-php>

[http://php.net/manual/pt\\_BR/function.setcookie.php](http://php.net/manual/pt_BR/function.setcookie.php)

[http://php.net/manual/pt\\_BR/function.include.php](http://php.net/manual/pt_BR/function.include.php)