



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
CAMPUS PARNAÍBA

PHP: Sessões, Cookies, Cabeçalhos HTTP, Autenticação e Autorização

Programação para WEB

www.ifpi.edu.br



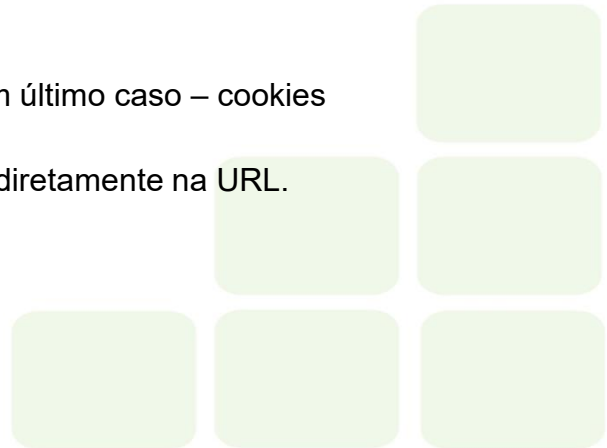
Sessões

- Permitem armazenar informações importantes do usuário do site e acessá-las em qualquer página por determinado período de tempo.
- Servem para armazenamento de dados no servidor.
- Pode ser importante armazenar as informações do usuário, como sua identificação de login, seu carrinho de compras, produtos visitados, etc.
- Um arquivo no servidor é criado por padrão no diretório temporário, com o nome do arquivo iniciando por "sess_" e seguido do ID da sessão:
 - Arquivo: sess_4e2eca1a2b20237ce861d690c68294cf
 - Caminho: C:\xampp\tmp No Ubuntu: /var/lib/php5/



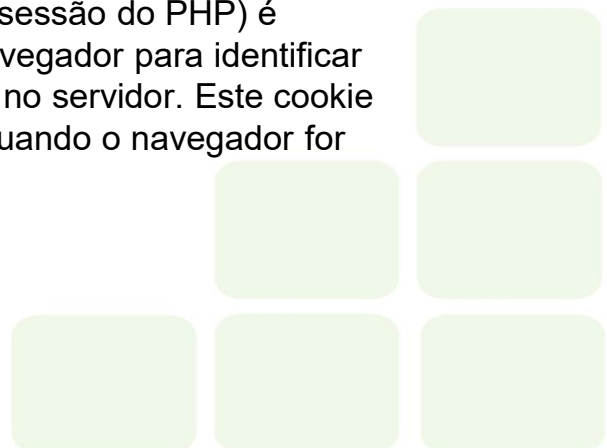
ID de Sessão

- Há 2 formas de passar o ID de sessão para o servidor:
 - Via Cookie (Recomendada)
 - Via parâmetro de URL (só em último caso – cookies desabilitados no navegador)
 - O ID de sessão é inserido diretamente na URL.



ID de Sessão via Cookie

- Ao fazer uso de sessão em PHP, um cookie chamado PHPSESSID (identificador de sessão do PHP) é automaticamente criado no navegador para identificar que sessão está sendo usada no servidor. Este cookie é automaticamente excluído quando o navegador for fechado.





Iniciando uma Sessão

- Para iniciar uma nova sessão ou para abrir uma sessão já iniciada, deve-se usar a função **`session_start()`**.
 - **`session_start()`** OBRIGATORIAMENTE deve vir antes de qualquer saída HTML.



Armazenando algo na Sessão

- Opções:
 - Via variável **`$_SESSION`**
 - Via função **`session_register()`**





Armazenando algo na Sessão

```
<?php
session_start();
$variavel_01 = "Programação";
$variavel_02 = "para WEB";
session_register('variavel_01');
$_SESSION['variavel_02'] = $variavel_02;
?>
<html>
<head>
  <title>CEFET-PI PHB</title>
</head>
<body>
<p>Informações sobre a sessão:</p>
<?php
echo '<p>Session ID: ' . session_id() . '</p>';
print_r($_SESSION);
?>
</p>
</body>
</html>
```



Eliminando algo da Sessão

- session_unregister()
 - Elimina uma determinada variável da sessão.
- session_unset()
 - Elimina todas as variáveis da sessão.



Tempo de vida da Sessão

- **session_cache_expire()**
 - Retorna o tempo de expiração das sessões ou define novo tempo de expiração em minutos.
 - Para redefinir o tempo de expiração, deve ser executada antes de **session_start()**.
 - Tempo padrão de duração da sessão: 180 minutos
 - Para alterar o tempo padrão, deve-se chamar **session_cache_expire()** a cada página que faça uso da sessão.



Destruindo uma Sessão

- A destruição da sessão é feita através da função **session_destroy()**.
- Antes de chamar a função **session_destroy()**, deve-se primeiro abrir a sessão com **session_start()**.
- Essa função é normalmente utilizada quando um usuário requisita sua saída da aplicação (logout).



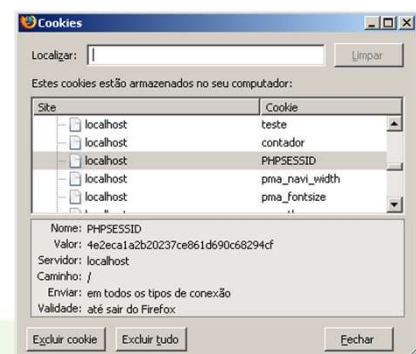
Cookie

- É uma pequena quantidade de informação armazenada na máquina cliente que possibilita a identificação de informações dos usuários que acessam o servidor.
- Pode estar na memória do navegador ou armazenada no disco.
- Contém um par: chave, valor.
- Usar com moderação.
- O uso de cookies pode permitir que o usuário após ter se autenticado, desligue o computador, acesse o site um tempo depois e não precise se autenticar novamente. Ex: GMail, Hotmail, Yahoo, etc.
 - Para utilizar esse recurso, o usuário aceita a opção “salvar as minhas informações neste computador”.



Informações contidas em um Cookie

- **Nome:** PHPSESSID
- **Valor:** 4e2eca1a2b20237ce861d690c68294cf
- **Servidor:** localhost
- **Caminho:** /
- **Enviar:** em todos os tipos de conexão
- **Validade:** até sair do Firefox





Cookie

- Os cookies são uma parte do cabeçalho HTTP, logo a função **setcookie()** precisa ser chamada antes que qualquer outro dado seja enviado ao navegador.
- Qualquer cookie enviado para o cliente automaticamente será incluído no array `$_COOKIE`.
- Para armazenar um array em cookies, acrescente `[]` ao nome do cookie.



Adicionando Cookies

```
<?php
/* dura até o fechamento do browser */
setcookie('meu_cookie1', 'PHP');

/* dura 1 hora */
setcookie('meu_cookie2', 'WEB', time()+3600);

/* dura 1 hora */
setcookie('meu_cookie3', 'NOW', time()+3600,
    '/regis/', 1);
?>
```



Usando Cookies

```
<?php
echo 'meu_cookie1: ';
echo $_COOKIE['meu_cookie1'].'<br/>';
echo 'meu_cookie2: ';
echo $_COOKIE['meu_cookie2'].'<br/>';
echo 'meu_cookie3: ';
echo $_COOKIE['meu_cookie3'].'<br/>';
print_r($_COOKIE);
?>
```



Removendo Cookies

```
<?php
// configurar valor com string vazia
setcookie ('meu_cookie1', '');
setcookie ('meu_cookie2', '');

// configurar a data de expiração para o passado
setcookie ('meu_cookie3', '', time() - 3600);
?>
```




Adicionando Array em Cookies

```
<?php
setcookie('pessoa[0]', 'Maria');
setcookie('pessoa[1]', 'João');
setcookie('pessoa[2]', 'José');
?>
```



Contador (Session)

```
<?php
    session_start();
    $_SESSION['contador_session']++;
?>
<html>
<head>
    <title>Contador</title>
</head>
<body>
<p>Contador Session:
<?php echo $_SESSION['contador_session'] ?></p>
</body>
</html>
```



Contador (Cookie)

```
<?php
    setcookie('contador_cookie',
        $_COOKIE['contador_cookie'] + 1);
?>
<html>
<head>
    <title>Contador</title>
</head>
<body>
<p>Contador Cookie:
<?php echo $_COOKIE['contador_cookie'] ?></p>
</body>
</html>
```



Inserindo itens em um array – usando Session

```
<?php
session_start();
if (strlen(trim($_POST['nome'])) > 0) {
    if (gettype($_SESSION['nomes']) == 'array') {
        array_push($_SESSION['nomes'], $_POST['nome']);
    } else {
        $_SESSION['nomes'] = array($_POST['nome']);
    }
}
?>
<html>
<head><title>Array usando Session</title></head>
<body>
    <form action="session_array.php" method="post">
        <p>Nome: <input name="nome" type="text"/></p>
        <input type="submit" value="Inserir Nome"/>
    </form>
    <?php
    if (isset($_SESSION['nomes'])) {
        echo '<p>Nomes inseridos na sessão:</p>';
        foreach ($_SESSION['nomes'] as $item) {
            echo $item.<br/>';
        }
    }
    ?>
</body>
</html>
```



Inserindo itens em um array – usando Cookies

```
<?php
if (strlen(trim($_POST['nome'])) > 0) {
    $tamanho = count($_COOKIE['nomes']);
    setcookie("nomes[$tamanho]", $_POST['nome']);
}
?>
<html>
<head><title>Array usando Cookies</title></head>
<body>
    <form action="cookies_array.php" method="post">
        <p>Nome: <input name="nome" type="text"/></p>
        <input type="submit" value="Inserir Nome"/>
    </form>
<?php
if (isset($_COOKIE['nomes'])) {
    echo '<p>Nomes inseridos no cookie:</p>';
    foreach ($_COOKIE['nomes'] as $item) {
        echo $item.'<br/>';
    }
    echo $_POST['nome'].'<br/>';
}
?>
</body>
</html>
```



Cabeçalhos HTTP

- Redirecionamento de página
 - A função exit interrompe a execução do script e pode mostrar uma mensagem.

```
<?php
header('Location: http://localhost/xampp');
exit;
?>
```

```
<?php
$servidor = 'http://' . $_SERVER['HTTP_HOST'];
header('Location: ' . $servidor . '/xampp/');
exit;
?>
```



Redirecionamento via HTML

```
<html>
  <head>
    <meta http-equiv="refresh"
      content="0;url=http://localhost/xampp/" />
  </head>
  <body>
    <p>Redirecionando...</p>
  </body>
</html>
```



Itens via Cookies - Exibição

```
<!-- array_cookies.php -->
<html>
  <head><title>Array usando Cookies</title></head>
  <body>
    <form action="array_cookies_processa.php" method="post">
      <p>Nome: <input name="nome" type="text"/></p>
      <input type="submit" value="Inserir Nome"/>
    </form>
    <?php
    if (isset($_COOKIE['nomes'])) {
      echo '<p>Nomes inseridos no cookie:</p>';
      $i = 1;
      foreach ($_COOKIE['nomes'] as $item) {
        echo "$i. $item <a
          href=\"array_cookies_processa.php? remover=$item\">
            Remover</a><br/>";
        $i++;
      }
    }
    ?>
  </body>
</html>
```



Itens via Cookies - Processamento

```
<!-- array_cookies_processa.php -->
<?php
    if (isset($_GET[remover])) {
        $posicao = array_search($_GET[remover], $_COOKIE['nomes']);
        setcookie("nomes[$posicao]", '');
        header('Location: array_cookies.php');
        exit;
    } else if (strlen(trim($_POST['nome'])) > 0) {
        end($_COOKIE['nomes']);
        $pos_final = key($_COOKIE['nomes']) + 1;
        setcookie("nomes[$pos_final]", $_POST['nome']);
        header('Location: array_cookies.php');
        exit;
    }
?>
```



Autenticação e Autorização

● Autenticação

- É o processo de identificação dos usuários do aplicativo.
- Normalmente essa identificação é determinada pelo par usuário e senha, mas também pode ser feita através de biometria, etc.

● Autorização

- Controla as permissões de acesso de usuários autenticados a recursos do sistema.
- Determina se um usuário autenticado pode ou não acessar um determinado recurso.



Autenticação HTTP

- O protocolo HTTP permite o pedido de usuário e senha via janela pop-up.
- Isso é feito através do cabeçalho WWW-Authenticate.
- Após a autenticação, as seguintes variáveis são criadas:
 - `$_SERVER['PHP_AUTH_USER']` – Usuário
 - `$_SERVER['PHP_AUTH_PW']` – Senha
 - `$_SERVER['PHP_AUTH_TYPE']` – Basic



Autenticação HTTP

```
<?php
function authenticate() {
    header('WWW-Authenticate: Basic realm="Login"');
    header('HTTP/1.0 401 Unauthorized');
    echo 'Usuário ou senha invalido(s).';
    exit;
}
if (!isset($_SERVER['PHP_AUTH_USER'])) {
    authenticate();
}
if ($_SERVER['PHP_AUTH_USER'] != 'regis' ||
    $_SERVER['PHP_AUTH_PW'] != '1234') {
    authenticate();
}
?>
<html>
<head><title>Página inicial</title></head>
<body>
<p>Bem-vindo ao nosso site!!!</p>
</body>
</html>
```



Autenticação – index.php

```
<?php
    session_start();
    $usuario = $_SESSION[usuario];
?>
<html>
    <head>
        <meta http-equiv="content-type"
              content="text/html; charset=UTF-8" />
        <title>Página inicial</title>
    </head>
    <body>
        <h1>Nossa aplicação</h1>
        <?php if ($usuario) { ?>
        <p><?php echo $usuario; ?>, seja bem-vindo ao nosso
          site. Clique <a href="sair.php">aqui</a>
          para sair.</p>
        <?php } else { ?>
        <p>É necessário fazer <a href="login.php">Login
          </a> para acessar o sistema.</p>
        <?php } ?>
    </body>
</html>
```



Autenticação – login.php

```
<?php
    session_start();
    $usuario = $_POST[usuario];
    $senha = $_POST[senha];
    if ($usuario == 'regis' && $senha == '12345') {
        $_SESSION[usuario] = $usuario;
        header('Location: index.php');
        exit;
    } else {
        if ($usuario && $senha)
            $mensagem = 'Usuário e/ou senha inválido(s). Tente novamente...';
    }
?>
<html>
    <head>
        <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
        <title>Página de Login</title>
    </head>
    <body>
        <h1>Login</h1>
        <?php if ($mensagem) {
            echo "<p>$mensagem</p>";
        } ?>
        <form action="login.php" method="POST">
            <p>Usuário: <input type="text" name="usuario" /></p>
            <p>Senha: <input type="password" name="senha" /></p>
            <p><input type="submit" />
        </form>
    </body>
</html>
```



Autenticação – sair.php

```
<?php
    session_start();
    session_destroy();
    header('Location: index.php');
?>
```

