

# PHP

## Sesiones

# Sesiones

El soporte para sesiones en PHP consiste en una forma de preservar cierta información a través de accesos subsiguientes. Esto habilita la construcción de aplicaciones más personalizadas e incrementa el atractivo de un sitio web.

# Sesiones

Básicamente una sesión es la secuencia de páginas que un usuario visita en un sitio web, desde que entra en nuestro sitio, hasta que lo abandona.

# Sesiones

El término sesión en PHP (session en inglés) se aplica a la secuencia de navegación, para ello se crea un identificador único y se asigna a cada una de las sesiones de navegación. A este identificador se le denomina como la sesión.

# Iniciar sesión

Para utilizar sesiones en PHP lo primero que se debe realizar es iniciarlas. Se puede hacer explícitamente mediante la función `session_start()` o al registrar una variable en una sesión mediante `session_register('miVariable')`. En ambos casos se crea una sesión si no existe o se retoma la sesión actual.

# ej1.php

```
<?php
    session_start();
    echo "Se ha iniciado la sesión";
?>
```

Para el ejemplo anterior, si el usuario tiene las cookies activadas, PHP inserta de forma automática la sesión y ésta será enviada de una página a otra.

Es posible obtener en cualquier momento la sesión mediante la función `session_id()`.



ej2.php

```
<?php
    session_start();
    echo "ID de la sección actual: ".session_id();
?>
```

# Registro de variables

Es posible registrar una variable en la sesión mediante la función `session_register()`

# ej3.php

```
<?php
    // Se registra la variable contador en la
    sesión
    session_register('contador');

    /* Enlace hacia la propia página que
    incrementa la variable contador */
    echo '<a
href="'. $_SERVER['PHP_SELF'] . '?' . SID . '">Cont
ador: ' . ++$_SESSION['contador'] . '</a>';
?>
```

`$_SERVER['PHP_SELF']` es una variable reservada que hace referencia al propio script en ejecución (constante de PHP4). `SID` que contiene el nombre de la sesión y el identificador de la misma.

# Nombre de sesión

Es posible personalizar la sesión tanto como se desee. Existe la variable `session_name()` que permite insertar un nombre a la sesión iniciada.

# ej4.php

```
<?php
    session_name('misesion');
    session_register('contador');
    echo '<a
href="'.$_SERVER['PHP_SELF'].'?'.SID.'">
        Contador vale:
'.++$_SESSION['contador'].'
        </a><br>';
    echo 'El nombre de la sesi&ocute;n es:
'.session_name();
    echo '<br>La sesi&ocute;n es'
```

El nombre de la sesión se debe asignar antes que ninguna otra función, incluyendo `session_start()`. Sin embargo, no se debe olvidar iniciar la sesión.

# Carrito de compras electrónico

Abriendo una sesión y utilizando la instrucción `$_SERVER['PHP_SELF'].'?'.SID` es posible actualizar la página y, con ello, toda la información que contiene la sesión abierta. Esto permite actualizar información (caso del contador) o almacenar nueva.



Los elementos que se adjuntan a la sesión se pueden obtener dependiendo del modo o método de envío.

Si se realiza una solicitud (como agregar un nombre a la sesión) el elemento se puede recuperar a partir del arreglo `$_REQUEST`.

Si se envía un formulario el elemento se puede recuperar dependiendo del método elegido mediante los arreglos `$_GET` o `$_POST`.

Si se registra una variable o se ingresa, explícitamente, un elemento en el arreglo `session`, se puede recuperar la información con el arreglo `$_SESSION`.

# carrito

```
<?php
    session_start();
    $item=$_POST['item'];
    $cantidad=$_POST['cantidad'];

    $itemsEnCesta=$_SESSION['itemsEnCesta'];
    $encontrado=0;
```

# carrito

```
if ($item){  
    if (!isset($itemsEnCesta)){  
        $itemsEnCesta[$item]=$cantidad;  
    }else{  
        foreach($itemsEnCesta as $k =>  
$v){  
            if ($item==$k){  
  
$itemsEnCesta[$k]+=$cantidad;  
                $encontrado=1;  
            }  
        }  
    }  
}
```

# carrito

```
<html>
```

```
<body>
```

```
  <tt>
```

```
    <form action="<?php  
$_SERVER['PHP_SELF']. "?" . $SID ?>"  
method="post">
```

```
      Nombre del producto <input  
type="text" name="item" size="20"><br>
```

```
      Número de unidades <input  
type="text" name="cantidad" size="20"><br>
```

```
      <input type="submit" value="Añadir a
```

Es posible eliminar un elemento de un arreglo mediante la función unset. unset recibe como parámetro el arreglo con el índice a eliminar. Su sintaxis es la siguiente:

```
unset($sistemas['Microsoft']);
```

# Validación de usuarios

Las sesiones son una excelente herramienta para validar que un usuario se ha registrado e ingresado al sistema de manera correcta.

# Ejemplo login



# Ejercicio

A partir del ejemplo anterior, crear los cambios necesarios para que cuando se seleccione el enlace de la Conjetura de Ulam, ésta página y las subsecuentes validen que el usuario sea válido.

# Ejercicio

Hacer los cambios pertinentes para que el login de usuario se lleve a cabo a partir de los usuarios existentes en una base de datos.

Una vez logeado el usuario, éste puede agregar o quitar elementos en un carrito de compras. Al final se debe mostrar la lista de productos comprados, la cantidad, el precio por producto y el total de la compra.