

Tema 8. SESSIONS EN PHP

Fins ara hem vist 3 formes bàsiques de fer que la informació generada en un script estigui disponible en scripts diferents al de creació: utilitzant formularis, passant les variables i els seus valors mitjançant la URL, o bé, definint cookies.

Aquests mètodes no són molt útils quan els scripts que han de compartir la informació no s'executen de forma seqüencial, o bé, estan distants els uns dels altres dins del fluxe normal d'execució de l'aplicació.

Encara que, en principi, l'ús de cookies sembla adequat per solucionar aquest problema (poden ser consultades en qualsevol moment fins a la seva eliminació del sistema client), s'ha de tenir en compte que no tots els navegadors tenen disponible aquesta funcionalitat i que, inclús tenint-la, aquesta pot estar deshabilitada per part de l'usuari.

Per solucionar aquesta situació, PHP proporciona les *variables de sessió*, que són variables que estaran disponibles a les diferents peticions al llarg d'un interval de temps que l'usuari necessiti fer ús de l'aplicació web. Són com variables globals a l'aplicació sencera ja que estaran accessibles per tots els programes de l'aplicació.

El cicle de vida d'una sessió comença a l'instant en que l'usuari es connecta a l'aplicació i acaba a l'instant en que surt d'aquesta, tanca el servidor, o bé, està una durada de temps (fixat pel sistema) sense treballar amb l'aplicació.

Totes les variables de sessió s'emmagatzemen al servidor, per tant, en un moment donat, es podran tenir moltes variables amb el mateix nom però amb continguts diferents. Per a evitar confusions, cada variable de sessió està vinculada a una única sessió/usuari, que s'identifica amb un identificador de sessió únic. La forma de mantenir de forma persistent aquest identificador al llarg de la sessió és utilitzant cookies o bé propagar-lo mitjançant la URL.

8.1. CREACIÓ DE SESSIONS

La configuració per defecte de PHP té deshabilitades les sessions; això vol dir que, quan es vulgui fer ús d'elles, s'han d'indicar de forma explícita. Canviant la configuració (modificant el valor de la directiva **session.auto_start** a 1 al fitxer **php.ini**), es poden activar per defecte les sessions en PHP de forma que en el primer accés de cada usuari al sistema se li crei una sessió.

La forma principal d'activar l'ús de sessions quan es vulgui es fa mitjançant l'ús de la funció **session_start()** amb la següent sintaxis:

```
bool session_start()
```

Aquesta funció crea una nova versió i genera el nou identificador, o retorna la sessió en el cas que ja existeixi, utilitzant l'identificador de sessió que s'han propagat fent ús de la URL o les cookies.

En el moment de crear una sessió es genera un identificador únic per a ella. Fent ús de la funció **session_id()**, es pot recuperar o modificar aquest valor. La seva sintaxis és:

```
string session_id([string id])
```

Si es crida a la funció sense fer ús del paràmetre `id`, s'obté una cadena amb l'identificador de la sessió actual (s'ha de cridar una vegada existeixi la sessió). En cas contrari, fen ús del paràmetre `id`, s'està reemplaçant l'identificador de la sessió actual pel valor donat per paràmetre (s'ha de cridar abans de crear la sessió).

També és possible identificar les sessions assignant-li un nom específic. Per a això s'utilitzarà la funció `session_name()` amb la sintaxis:

```
string session_name([string nom])
```

Si no es fa ús del paràmetre `nom`, s'obté el nom de la sessió actual; en cas contrari, si s'informa aquest paràmetre, la sessió actual passarà a tenir per nom l'argument donat.

8.2. ACCÉS A LES VARIABLES DE SESSIÓ

La forma d'accés a les variables de sessió és mitjançant l'array associatiu `$_SESSION` disponible com a variable global (`$HTTP_SESSIONS_VARS` a les versions anteriors a la 4.2.0).

En el següent exemple es mostra la creació d'una sessió i d'una variable de sessió, a més de l'accés, actualització i eliminació per poder obtenir un comptador d'accessos.

```
<?php
    session_start();
    if (isset($_SESSION['contador'])) {
        $_SESSION['contador']++;
    }
    else {
        $_SESSION['contador'] = 0;
    }
    $nombre_anterior = session_name('SESION_CONTADOR');
?>
<HTML>
<HEAD>
    <TITLE>Treballant amb Sessions</TITLE>
</HEAD>
<BODY>
    <CENTER>
        <H2>Treballant amb Sessions</H2>
        <TABLE BORDER="1" CELLPADDING="2" CELLSPACING="4">
            <TR ALIGN="center" BGCOLOR="orange">
                <TD COLSPAN="2"><B>Informació de la Sessió</B></TD>
            </TR>
            <TR>
                <TD BGCOLOR="orange">ID</TD>
                <TD><?php echo session_id() ?></TD></TR>
            <TR>
                <TD BGCOLOR="orange">Nombre d'accessos</TD>
                <TD><?php echo $_SESSION['contador'] ?></TD></TR>
            <TR>
                <TD BGCOLOR="orange">Nom actual</TD>
                <TD><?php echo session_name() ?></TD></TR>
            <TR>
                <TD BGCOLOR="orange">Nom anterior</TD>
```

```

        <TD><?php echo $nombre_anterior ?></TD></TR>
    </TABLE>
    <BR><PRE>
        <A HREF="sesiones1.php">Actualitzar</A> |
                                <A HREF="sesiones2.php">Resetejar comptador</A>

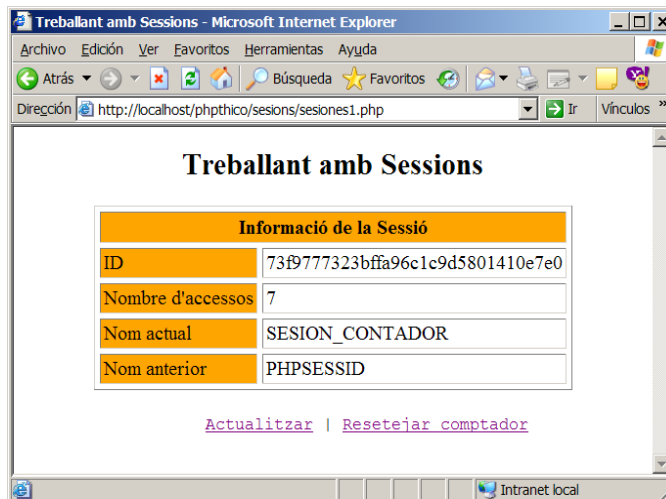
    </PRE>
</CENTER>
</BODY>
</HTML>

```

Com es pot veure, per crear una nova variable de sessió, només s'ha de definir dins de l'array **\$_SESSION**.

En el codi es pregunta si la variable de sessió té valor (si està definida) i, d'aquesta forma, saber si s'ha de crear o actualitzar el valor. A més s'obté l'identificador de sessió i modifica el nom assignat per defecte.

A la següent imatge es mostra el resultat després d'haver realitzat 7 accessos a la pàgina que conté el comptador:



Al segon script d'aquest exemple s'esborra la variable de sessió creada al script anterior:

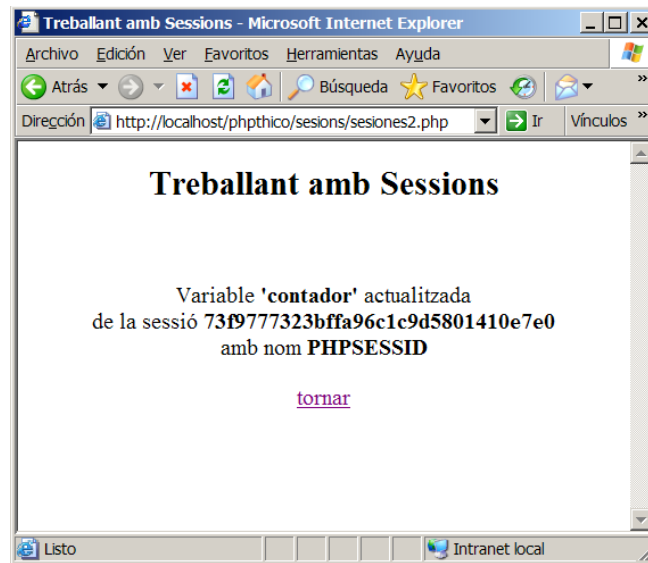
```

<?php
    session_start();
    unset($_SESSION['contador']);
?>
<HTML>

<HEAD>
    <TITLE>Treballant amb Sessions</TITLE>
</HEAD>
<BODY>
    <CENTER>
        <H2>Treballant amb Sessions</H2><BR><BR>
        <P>Variable <B>'contador'</B> actualitzada<BR>
        de la sessió <B><?php echo session_id() ?></B><BR>
        amb nom <B><?php echo session_name() ?></B>
        <BR><BR><A HREF="sesiones1.php">tornar</A></P>
    </CENTER>
</BODY>
</HTML>

```

Com es pot observar, la sessió segueix sent la mateixa (el seu identificador és el mateix), però ha perdut el nom de la sessió. Aquest valor és necessari actualitzar-lo en cada petició; si no, es restaura amb el valor emmagatzemat per defecte a la directiva **session.name** del fitxer de configuració **php.ini**:



Existeix una forma alternativa per accedir a les variables de sessió sense utilitzar la variable global `$_SESSION` que ens permet manegar aquestes variables directament fent ús del seu nom. Per a que aquesta opció estigui disponible, serà necessari activar la directiva **register_globals** en el fitxer de configuració **php.ini**.

NOTA: per a qüestions de seguretat i rendiment és preferible fer ús de l'array `$_SESSION`.

Aquesta forma de treballar està associada a 3 funcions PHP:

- `session_register(nom [,nom])`: registra una nova variable global per a la sessió actual. En cas de que no existeixi una sessió, la crea realitzant una crida implícita a la funció **session_start()**. Pot rebre múltiples arguments, cadascun dels quals serà una nova variable global per a la sessió actual (retorna *true* quan totes les variables passades com a argument han estat creades sense cap problema).
- `session_is_registered(nom)`: retorna *true* quan la variable que s'ha passat com a argument a la crida està registrada a la sessió actual.
- `session_unregister(nom)`: evita que la variable passada com a argument a la crida sigui salvada com a part de la sessió actual, és a dir, l'elimina de la sessió, però segueix sent una variable de l'script on s'executa.

El següent codi mostra el mateix exemple que l'anterior, però fent ús d'aquest mètode d'accés a les variables de sessió:

```
<?php
session_start();
if (!session_is_registered('contador')) {
    session_register('contador');
    $contador = 0;
}
else {
    $contador++;
}
$nombre_anterior = session_name('SESION_CONTADOR');
?>
<HTML>
<HEAD>
<TITLE>Treballant amb Sessions</TITLE>
</HEAD>
<BODY>
<CENTER>
<H2>Treballant amb Sessions</H2>
<TABLE BORDER="1" CELLPADDING="2" CELLSPACING="4">
<TR ALIGN="center" BGCOLOR="orange">
<TD COLSPAN="2"><B>Informació de la Sessió</B></TD>
</TR>
<TR>
<TD BGCOLOR="orange">ID</TD>
<TD><?php echo session_id() ?></TD>
</TR>
<TR>
<TD BGCOLOR="orange">Nombre d'accessos</TD>
<TD><?php echo $contador ?></TD>
</TR>
<TR>
<TD BGCOLOR="orange">Nom actual</TD>
<TD><?php echo session_name() ?></TD>
</TR>
<TR>
<TD BGCOLOR="orange">Nom anterior</TD>
<TD><?php echo $nombre_anterior ?></TD>
</TR>
</TABLE>
<BR><PRE>
<A HREF="sesiones3.php">Actualitzar</A> |
<A HREF="sesiones4.php">Resetejar comptador</A>
</PRE>
</CENTER>
</BODY>
</HTML>
```

El fitxer encarregat d'eliminar la variable de la sessió té el següent contingut:

```
<?php
  session_start();
  if(session_is_registered('contador'))
    session_unregister('contador');
?>
<HTML>
<HEAD>
  <TITLE>Treballant amb Sessions</TITLE>
</HEAD>
<BODY>
  <CENTER>
    <H2>Treballant amb Sessions</H2><BR><BR>
    <P>Variable <B>'contador'</B> actualitzada<BR>
      de la sessió <B><?php echo session_id() ?></B><BR>
      amb nom <B><?php echo session_name() ?></B>
      <BR><BR><A HREF="sesiones3.php">tornar</A></P>
  </CENTER>
</BODY>
</HTML>
```

Les finestres resultants són iguals als de l'exemple anterior.

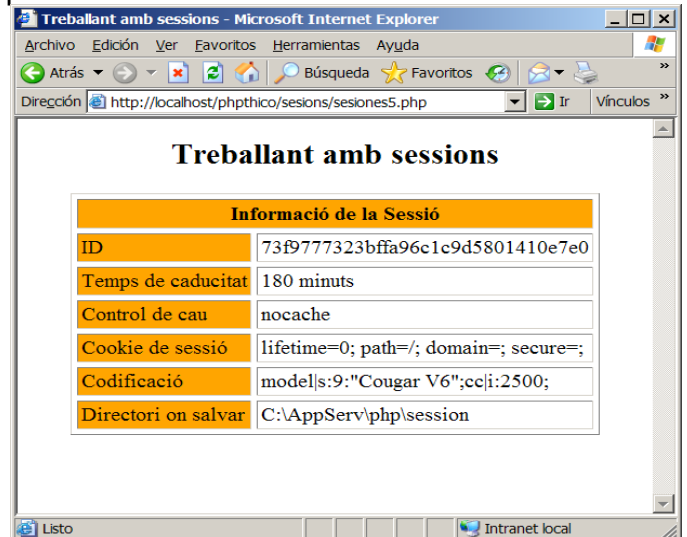
8.3. ALTRES FUNCIONS ASSOCIADES AL MANEGAMENT DE FUNCIONS

Podem veure les funcions més representatives que PHP proporciona per al treball amb sessions:

- **`session_destroy()`**: elimina totes les dades associades amb la sessió actual, sense modificar les variables globals associades a la sessió ni la cookie de sessió.
- **`session_unset()`**: allibera tots els recursos associats a les variables de sessió actualment registrades.
- **`session_encode()`**: codifica les dades de la sessió actual en una cadena.
- **`session_encode(dades)`**: decodifica les dades d'una sessió passades com a argument en una cadena, generant les variables guardades en aquesta sessió.
- **`session_cache_expire([caducitat])`**: retorna el temps que resta en minuts per a que la sessió finalitzi. Si es passa un valor enter com a argument en la crida de la funció, s'actualitzarà el temps de caducitat de la sessió.
- **`session_get_cookie_params()`**: obté els paràmetres de la cookie de sessió (durada, camí, domini i seguretat), en cas de que aquest hagi sigut el mètode escollit per gestionar les sessions.
- **`session_set_cookie_params(temps [,camí [,domini [,seguretat]])`**: ens permet canviar els paràmetres de la cookie de sessió actual.
- **`session_save_path([string camí])`**: permet obtenir o modificar la ruta on es guarden les dades de la sessió actual.

El següent exemple mostra la utilització d'aquestes funcions:

```
<?php
    session_start();
    $_SESSION['model'] = "Cougar V6";
    $_SESSION['cc'] = 2500;
?>
<HTML>
<HEAD>
    <TITLE>Treballant amb sessions</TITLE>
</HEAD>
<BODY>
    <CENTER>
        <H2>Treballant amb sessions</H2>
        <TABLE BORDER="1" CELLPADDING="2" CELLSPACING="4">
            <TR ALIGN="center" BGCOLOR="orange">
                <TD COLSPAN="2"><B>Informació de la Sessió</B></TD>
            </TR>
            <TR>
                <TD BGCOLOR="orange">ID</TD>
                <TD><?php echo session_id() ?></TD>
            </TR>
            <TR>
                <TD BGCOLOR="orange">Temps de caducitat</TD>
                <TD><?php echo session_cache_expire()," minuts" ?></TD>
            </TR>
            <TR>
                <TD BGCOLOR="orange">Control de cau</TD>
                <TD><?php echo session_cache_limiter() ?></TD>
            </TR>
            <TR>
                <TD BGCOLOR="orange">Cookie de sessió</TD>
                <TD>
                    <?php
                        $parametros=session_get_cookie_params();
                        while(list($par,$valor)=each($parametros)){
                            echo $par,"=",$valor," ";
                        }
                    ?>
                </TD>
            </TR>
            <TR>
                <TD BGCOLOR="orange">Codificació</TD>
                <TD><?php echo session_encode() ?></TD>
            </TR>
            <TR>
                <TD BGCOLOR="orange">Directorio on salvar</TD>
                <TD><?php echo session_save_path() ?></TD>
            </TR>
        </TABLE>
    </CENTER>
</BODY>
</HTML>
```



8.4. PARÀMETRES DE CONFIGURACIÓ DE SESSIONS

A la següent taula es mostra un resum de les directives del fitxer **php.ini** relacionades amb la gestió de sessions:

Directiva	Valor per defecte	Descripció
session.auto_start	0	Especifica si el mòdul que gestiona les sessions s'inicia automàticament en rebre una petició.
session.name	PHPSESSID	Especifica el nom de la sessió que s'utilitza com a nom de la cookie.
session.save_handler	files	Defineix el tipus de controlador que s'utilitza per emmagatzemar i recuperar les dades associades a la sessió. Es pot especificar <i>user</i> si les operacions les implementa l'usuari.
session.save_path	/tmp	Ruta on s'emmagatzemen les dades associades a les cookies.
session.serialize_handler	php	Defineix el controlador que s'utilitza per a guardar i restaruar les dades de forma serialitzada.
session.use_cookies	1	Indica si el mòdul pot utilitzar cookies per a guardar l'identificador de sessió en el costat del client.
session.use_only_cookies	0	Especifica si només s'han d'utilitzar cookies per a guardar l'identificatiu de sessió en el costat del client.
session.cookie_lifetime	0	Especifica la durada de la cookie en segons que s'envia al navegador. El valor 0 indica "fins que es tanqui el navegador"
session.cookie_path	/	Especifica la ruta a col·locar en <i>session_cookie</i> .
session.cookie_domain		Especifica el domini a establir en <i>session_cookie</i> .
session.referer_check		Conté la subcadena de comprovació de cada "HTTP referer".
session.cache_limiter	nocache	Especifica el mètode de control del cau a utilitzar a les pàgines de la sessió (<i>none, nocache, private, public, private_no_expire</i>).
session.cache_expire	180	Especifica el temps de vida en minuts de les pàgines de la sessió que es troben en cau.