

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

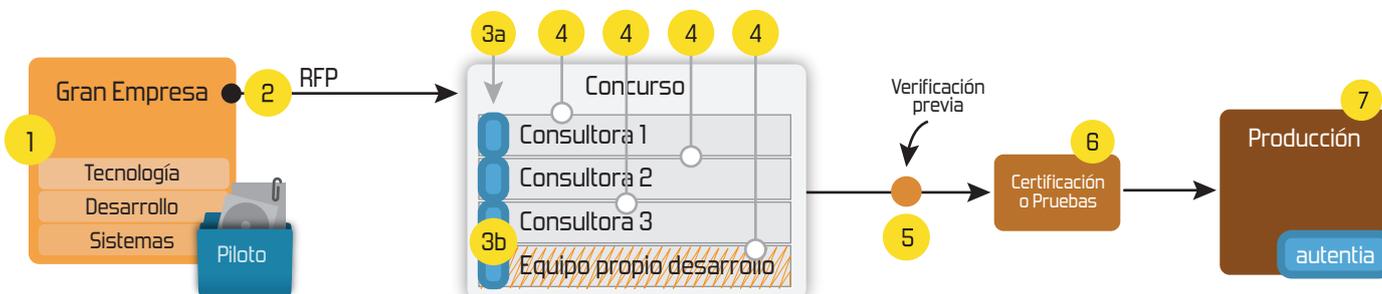
1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

Tutorial desarrollado por: Alejandro Perez García 2003-2005, nuestro experto en J2EE, Linux y optimización de aplicaciones empresariales.

Si te gusta lo que ves, **puedes contratarme** para impartir **cursos presenciales** en tu empresa o para ayudarte en proyectos (Madrid).

Contacta: alejandropg@autentia.com.



Descargar este documento en formato PDF [php_on_debian.pdf](#)

Primer Trabajo

El trabajo que buscas está aquí. Miles de ofertas de empleo
www.laboris.net

CCS

Centro de Cálculo de Sabadell 1ª empresa española de software
www.ccs.biz

Nexedi ERP5

Consulting and development services Open Source ERP5, CRM, CPS, Zope
www.nexedi.com

Anuncios Goooooogle

Anunciarse en este sitio

Como instalar MySQL, Apache y PHP en GNU / Linux (Debian)

1. Introducción

Con este tutorial se pretende montar un entorno de trabajo para desarrollar páginas en PHP.

Vamos a instalar:

- MySQL 4.1.1
- MySQL Control Center 0.9.4
- Apache 2.0.48
- PHP 4.3.4

La mayoría de las opciones de configuración se toman por defecto. Si el lector quiere más información sobre las posibilidades de configuración deberá remitirse a la documentación técnica.

Al final del tutorial se indican algunos enlaces donde se puede encontrar mucha más información sobre el tema.

2. Entorno

Hardware: Portátil Ahtex Signal X-9500M (Centrino 1.6 GHz, 1024 MB RAM, 60 GB HD).

Sistema Operativo: GNU / Linux, Debian Sid (unstable), Kernel 2.6.3, KDE 3.2

En primer lugar decir que esta configuración no está recomendada para entornos de producción. Si queremos instalar un entorno de producción deberemos usar Apache 1.x.

En segundo lugar indicar que entre los paquetes oficiales de Debian encontraremos paquetes para instalar Apache 2 y paquetes para instalar PHP 4, pero estos últimos dependen de Apache 1. Es decir, no podemos usar apt-get para realizar la instalación. Para solucionar esto tenemos tres opciones:

1. Esperar que saquen los paquetes con las dependencias correctas.
2. Buscar los paquetes en fuentes no oficiales.
3. Compilar nosotros mismos tanto Apache 2 como PHP 4.

En este tutorial veremos la tercera opción.

3. MySQL

Entre los paquetes oficiales de Debian no encontraremos la versión 4.1, así que nos vamos a la página de descargas de MySQL (<http://www.mysql.com/downloads/mysql-4.1.html>) y nos bajamos la versión 4.1.1-alpha, Standard para Linux (x86, libc6).

Para ello pinchamos sobre "Pick a mirror", elegimos la localización que más nos convenga, y esperamos a que se bajen las 22.6 MB del archivo mysql-standard-4.1.1-alpha-pc-linux-i686.tar.gz.

Para instalarlo no tenemos más que seguir los pasos que se describen en el fichero INSTALL-BINARY, que podemos encontrar dentro del propio archivo. Estos pasos son:

1. Creamos un usuario y un grupo para ejecutar MySQL.

```
# groupadd mysql
# useradd -g mysql mysql
```

2. Descomprimos el archivo mysql-standard-4.1.1-alpha-pc-linux-i686.tar.gz donde en /usr/local. Es aconsejable utilizar esta ruta, ya que es la localización por defecto. Si queremos instalarlo en otro sitio, luego tendríamos que cambiar ficheros de configuración. Para ello hacemos (suponiendo que el archivo que hemos descargado está en /download).

```
# cd /usr/local
# tar -xzf /download/mysql-standard-4.1.1-alpha-pc-linux-i686.tar.gz
```

3. Hacemos un link simbólico al directorio (para ahorrarnos el nombre tan largo ;)

```
# ln -s mysql-standard-4.1.1-alpha-pc-linux-i686/ mysql
```

4. Ejecutamos el script de instalación. Este script se encarga de crear las tablas internas de MySQL. Al ejecutar el script nos aparecerá un aviso para que cambiemos la clave del usuario root de MySQL. Los comandos que nos propone debemos ejecutarlos una vez esté arrancado MySQL (así que será mejor apuntarlos para no perderlos).

```
# cd mysql
# scripts/mysql_install_db
```

5. Ahora cambiamos el usuario y grupo de los ficheros.

```
# chown -R root .
# chown -R mysql data
# chgrp -R mysql .
```

6. Ya estamos listos para ejecutar MySQL.

```
# bin/mysql_safe --user=mysql &
```

7. Una vez está corriendo MySQL es el momento de ejecutar los comandos para cambiar la clave del usuario root de MySQL (se nombro en el paso).

Otra posibilidad para arrancar y parar el servidor es utilizar los comandos

```
# /usr/local/mysql/support-files/mysql.server start
# /usr/local/mysql/support-files/mysql.server stop
```

El script mysql.server puede usarse para arrancar el servidor en el arranque del sistema, copiándolo donde están los ficheros de arranque del sistema. En el caso de Debian deberíamos hacer:

```
# cp /usr/local/mysql/support-files/mysql.server /etc/init.d
```

Y ahora hacer un link en el run level que nos interese. Por ejemplo:

```
# cd /etc/rc2.d
# ln -s /etc/init.d/mysql.server S20mysql
```

También se puede usar el script mysql.server para parar el servidor cuando se pare el sistema. En Debían deberíamos hacer (suponiendo que ya hemos copiado el script en el directorio /etc/init.d):

```
# cd /etc/rc0.d
# ln -s /etc/init.d/mysql.server K20mysql
```

Al usar el script mysql.server, es conveniente, por motivos de seguridad, ejecutar MySQL como el usuario mysql (usuario que habíamos creado al principio), y no como el usuario root. Para ello debemos editar el script mysql.server, y cambiar la línea 148:

```
$bindir/mysql_d_safe --datadir=$datadir --pid-file=$pid_file &
```

por:

```
$bindir/mysql_d_safe --datadir=$datadir --pid-file=$pid_file --user=mysql &
```

4. MySQL Control Center

Esta es una herramienta de gran ayuda para la administración de MySQL, por lo que es altamente aconsejable su instalación.

Aunque la última versión es la 0.9.4 y la versión que podemos encontrar en los paquetes oficiales de Debian es esta misma, no podemos hacer la instalación a partir de los paquetes oficiales de Debian porque estos tienen una dependencia de mysql-common versión 4.0. Esto causa que si intentamos conectar con el servidor que acabamos de instalar obtendremos el siguiente mensaje de error:

```
[localhost] ERROR 1250: Client does not support authentication protocol requested by server; consider upgrading MySQL client
```

Es por esto que tendremos que descargar la versión Linux (x86, glibc 2.3) de <http://www.mysql.com/downloads/mysqlcc.html>.

Para ello, igual que antes, pinchamos sobre "Pick a mirror", elegimos la localización que más nos convenga, y esperamos a que se bajen las 4.3 MB del archivo mysqlcc-0.9.4-linux-glibc23.tar.gz.

En el fichero INSTALL.txt, que encontramos dentro del propio archivo, se describen los pasos para hacer la instalación. Estos pasos son:

1. Descomprimos el archivo mysqlcc-0.9.4-linux-glibc23.tar.gz donde queramos. Yo he utilizado /usr/local, igual que con el mysql. Para ello hacemos (suponiendo que el fichero que hemos descargado está en /download).

```
# cd /usr/local
# tar -xzf /download/mysqlcc-0.9.4-linux-glibc23.tar.gz
```

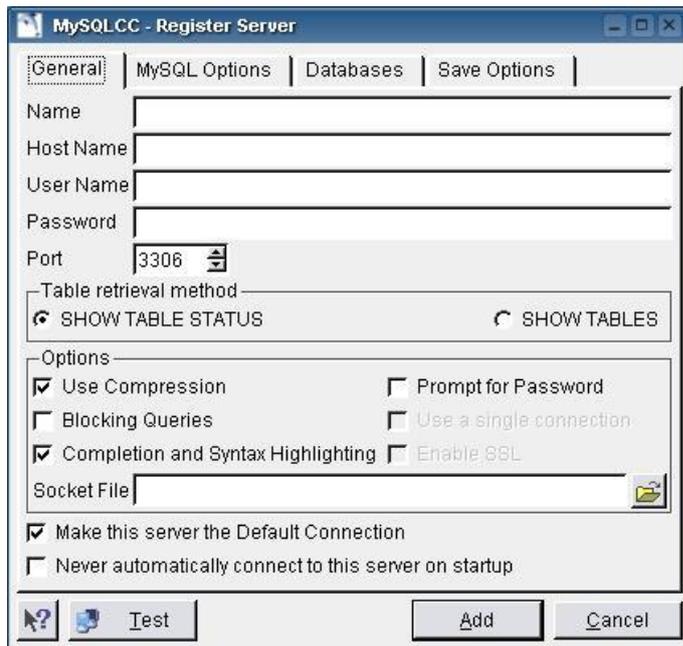
2. Hacemos un link simbólico al directorio (para ahorrarnos el nombre tan largo ;)

```
# ln -s mysqlcc-0.9.4-linux-glibc23.tar.gz/ mysqlcc
```

3. Ya estamos listos para ejecutar MySQL Control Center.

```
# cd mysqlcc
# ./mysqlcc
```

Deberíamos poder ver algo similar a:



En esta ventana debemos definir la conexión a MySQL. Debemos rellenar los campos:

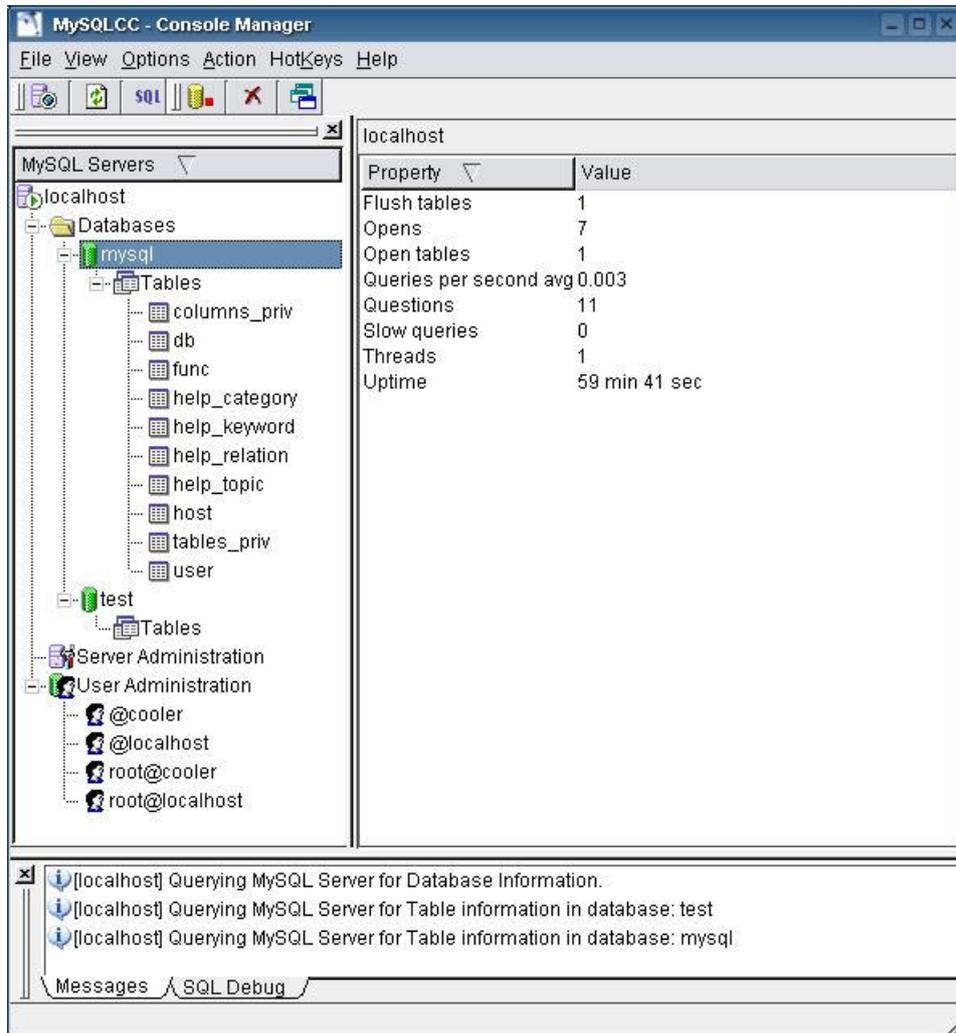
- Name: nombre que le queremos dar a la conexión.
- Host Name: nombre de la máquina donde está el servidor MySQL. En nuestro caso 'localhost'.
- User Name: nombre del usuario de MySQL para conectar con el servidor. Por ejemplo 'root'.
- Password: Clave que corresponde con el usuario del campo anterior.

Una vez rellenos los campos, la ventana debería quedar de una forma similar a:



Cuando terminemos de rellenar los campos, pulsamos el botón 'Add'.

Después, en la ventana que nos aparece, pulsamos con el botón derecho del ratón sobre el nombre de la conexión que acabamos de crear, y en el menú emergente pulsamos sobre 'Connect'. En este momento MySQL Control Center se conectará con el servidor, y si desplegamos las pestañas deberíamos ver algo similar a:



Ahora ya podemos administrar el servidor cómodamente.

5. Apache

Nos descargamos Apache 2.x. Para ello nos dirigimos a la página <http://httpd.apache.org/download.cgi>. Seleccionamos la localización que más nos convenga (si cambiamos la localización por defecto es necesario pulsar el botón 'Change'). Pinchamos sobre el enlace httpd-2.0.48.tar.gz. Esperamos a que se descargue el archivo.

Para instalarlo no tenemos más que seguir los pasos que se describen en los ficheros INSTALL y docs/manual/install.html.en (más detallado que el anterior), que podemos encontrar dentro del propio archivo. Estos pasos son:

1. Descomprimos el fichero `httpd-2.0.48.tar.gz`. Suponemos que el fichero está en `/download`, y como destino de la descompresión y para compilar usaremos el directorio `/usr/local/src`.

```
# cd /usr/local/src
# tar -zxf /download/httpd-2.0.48.tar.gz
```

2. Configuramos los fuentes para que la compilación se adapte a nuestro entorno y necesidades.

```
# cd httpd-2.0.48
# ./configure --prefix=/usr/local/apache2 --enable-so --with-mpm=prefork
```

Con esto compilaremos Apache con las opciones por defecto y especificamos:

`prefix`: indica dónde se instalará el Apache. Aunque `/usr/local/apache2` es la opción por defecto, lo especificamos para que quede claro. Si quisiéramos instalar Apache en otro sitio, tendríamos que indicarlo aquí.

enable-so: indica que queremos hacer carga dinámica de los módulos. Esto sirve para luego poder gargar el módulo de PHP.

with-mpm: mpm son las siglas de Multi-Processing Module. Esta opción sirve para especificar el módulo encargado del multiproceso (de como trabajara el Apache internamente). Aunque prefork es la opción por defeco para Linux, lo hemos puesto para que quede claro que es necesario usar este modo para que PHP funcione correctamente. Con el uso de otros modos no se asegura el correcto funcionamiento de PHP.

3. Compilamos los fuentes.

```
# make
```

4. Instalamos Apache:

```
# make install
```

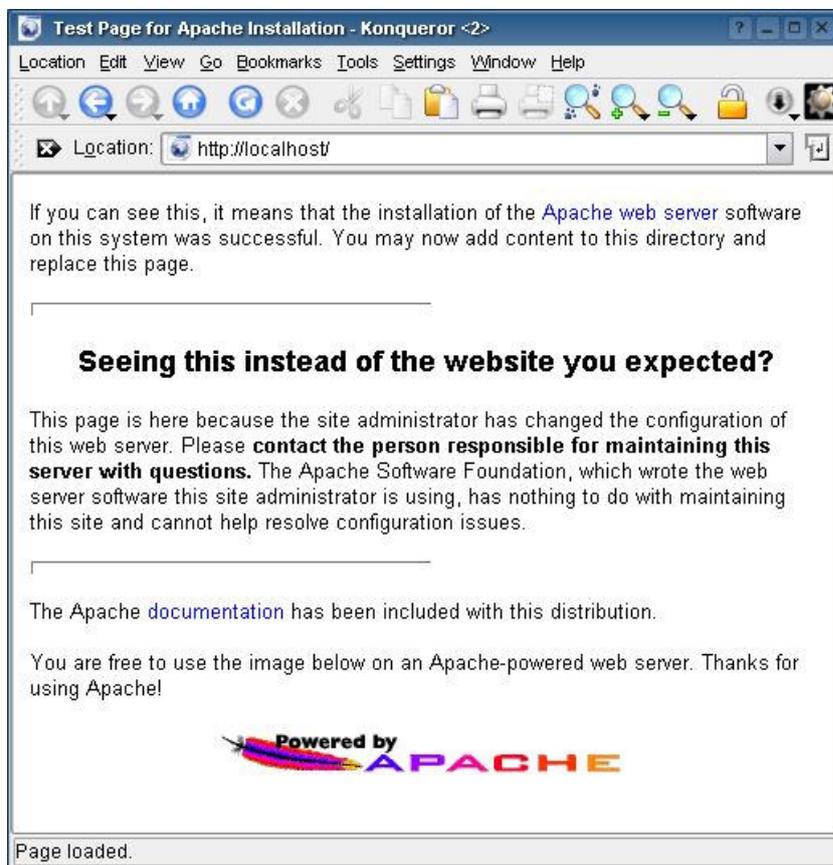
5. Levantamos Apache:

```
# cd /usr/local/apache2/bin
# apachectl start
```

Para comprobar que Apache funciona correctamente basta con ir con en nuestro explorador favorito a la dirección:

<http://localhost>

Deberíamos ver una página similar a:



Es importante parar le servidor para poder seguir con la instalación de PHP. Podemos usar el comando:

```
# apachectl stop
```

El script apachectl puede usarse para ejecutar el servidor en el arranque del sistema, copiándolo donde están los ficheros de arranque del sistema. En el caso de Debian deberíamos hacer:

```
# cp /usr/local/apache2/bin/apachectl /etc/init.d
```

Y ahora hacer un link en el run level que nos interese. Por ejemplo:

```
# cd /etc/rc2.d
# ln -s /etc/init.d/apachectl S20apachectl
```

También se puede usar el script mysql.server para parar el servidor cuando se pare el sistema. En Debían deberíamos hacer (suponiendo que ya hemos copiado el script en el directorio /etc/init.d):

```
# cd /etc/rc0.d
```

```
# ln -s /etc/init.d/apachectl K20apachectl
```

Ya sólo queda el último paso: la instalación de PHP.

6. PHP

Nos descargamos PHP 4.x. Para ello nos dirigimos a la página <http://www.php.net/downloads.php> y pinchamos sobre PHP 4.3.4 (tar.bz2 o tar.gz, el que más nos convenga; en mi caso he elegido el bz2 que es más pequeño).

Seleccionamos la localización que más nos interese y esperamos a que se descargue el archivo php-4.3.4.tar.bz2

Para instalarlo no tenemos más que seguir los pasos que se describen en el fichero INSTALL, que podemos encontrar dentro del propio archivo. Estos pasos son:

1. Igual que hemos hecho anteriormente, primero descomprimos el archivo.

```
# cd /usr/local/src
# tar --bzip2 -xf /download/php-4.3.4.tar.bz2
```

2. Configuramos los fuentes para su posterior compilación.

```
# cd php-4.3.4/
# ./configure --prefix=/usr/local/php4 --with-mysql=/usr/local/mysql --with-apxs2=/usr/local/apache2/bin/apxs
```

Con esto compilaremos Apache con las opciones por defecto y especificamos:

prefix: donde se instalará el php.

with-mysql: incluye soporte para MySQL. Le indicamos el directorio base donde habíamos instalado MySQL.

with-apxs2: compila el módulo para Apache 2. Le indicamos donde esta la utilidad de Apache: apxs. Es importante poner el 2, si no lo ponemos estaremos compilando el módulo para Apache 1, y no es lo que queremos.

3. Compilamos PHP.

```
# make
```

4. Instalamos PHP.

```
# make install
```

5. Copiamos el fichero de configuración de PHP, donde PHP espera encontrarlo. Esto es: el directorio que indicáramos con la opción prefix + /lib.

```
# cp php.ini-dist /usr/local/php4/lib/php.ini
```

6. Necesitamos que Apache reconozca los ficheros con extensión php. Para esto añadimos en el fichero /usr/local/apache2/conf/httpd.conf la línea (yo la he añadido en la línea 854):

```
AddType application/x-httpd-php .php
```

Ahora ya podemos iniciar Apache como se vio en el apartado anterior.

7. Comprobar que PHP funciona correctamente

Para comprobar si la instalación de PHP ha sido vamos ha hacer el típico ejemplo 'Hola Mundo', pero un poco más útil ;)

1. Con nuestro editor preferido creamos el fichero /usr/local/apache2/htdocs/hola_mundo.php

2. En su interior escribimos:

```
<?php
phpinfo();
?>
```

3. Ahora con nuestro explorador abrimos la dirección http://localhost/hola_mundo.php.

Si todo ha ido bien deberíamos ver algo similar a:



Se puede observar que la función `phpinfo()` es bastante útil, ya que nos permite ver bastante información del sistema.

8. Comprobar que PHP + MySQL funciona correctamente

Para hacer esta comprobación podemos cambiar el contenido del fichero `hola_mundo.php` por:

```
<?php
// Ejemplo de acceso a base de datos MySQL.

// Conseguimos la conexión.
$link = @mysql_connect('localhost', 'root', 'root');
if (!$link) {
    exit('<b>Error</b>: No se puede conectar con la base de datos. '.mysql_error());
}

$datab = 'mysql';
if (!@mysql_select_db($datab)) {
    exit('<b>Error</b>: No se encuentra la base de datos '$datab'.');
}

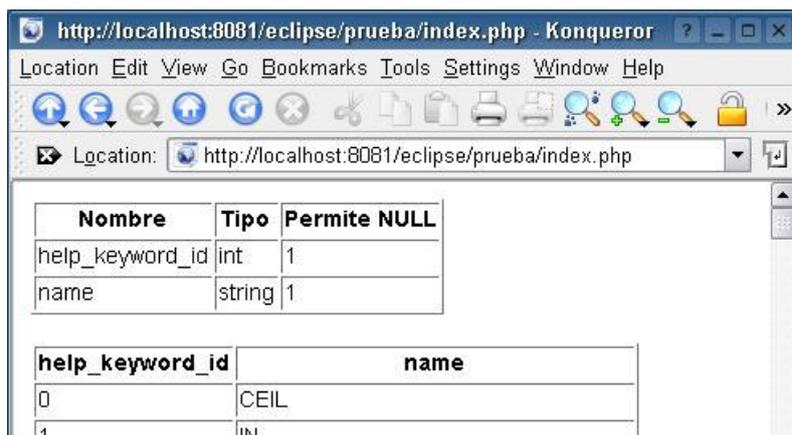
// Ejecutamos una consulta.
$sql = 'SELECT * FROM help_keyword';
$res = @mysql_query($sql);
if (!$res) {
    exit('<b>Error</b>: La consulta '$sql' ha fallado. '.mysql_error());
}

// Mostramos el metadata de la tabla.
echo '<table border="1">';
echo '<tr> <th>Nombre</th> <th>Tipo</th> <th>Permite NULL</th> </tr>';
$numFields = mysql_num_fields($res);
for ($i = 0; $i < $numFields; $i++) {
    $meta = mysql_fetch_field($res, $i);
    echo '<tr>';
    echo "&<td>$meta->name</td>";
    echo "<td>$meta->type</td>";
    echo "<td>$meta->not_null</td>";
    echo '</tr>';
}
echo '</table>';
echo '<br>';

// Mostramos el contenido de la tabla.
echo '<table border="1">';
echo '<tr>';
for ($i = 0; $i < $numFields; $i++) {
    $meta = mysql_fetch_field($res, $i);
    echo "&<th>$meta->name</th>";
}
echo '</tr>';
while ($row = mysql_fetch_row($res)) {
    echo '<tr>';
    foreach ($row as $col) {
        echo "&<td>$col</td>";
    }
    echo '</tr>';
}
echo '</table>';

// Liberamos los recursos y cerramos la conexión.
mysql_free_result($res);
mysql_close($link);
?>
```

Al ejecutar el fichero deberíamos conseguir un resultado similar a:



Nombre	Tipo	Permite NULL
help_keyword_id	int	1
name	string	1

help_keyword_id	name
0	CEIL
1	IN

9. Enlaces de interés

A continuación se listan unos enlaces donde se puede encontrar más información:

- MySQL: The World's Most Popular Open Source Database
<http://www.mysql.com/>
- Apache HTTP Server Version 2.0 Documentation
<http://httpd.apache.org/docs-2.0/>
- PHP: Servidores-Apache 2.0 – Manual
<http://www.php.net/manual/es/install.apache2.php>

10. Sobre el autor

Alejandro Pérez García

Dir. Implantación y Rendimiento

<mailto:alejandropg@autentia.com>

Autentia Real Business Solutions S.L.

<http://www.autentia.com>

Si desea contratar formación, consultoría o desarrollo de piezas a medida puede contactar con

Formación en nuevas tecnologías

[Autentia S.L.](#) Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto

[Soporte XML en Eclipse con X-MEN](#)

[Retoque fotográfico con GIMP](#)

[Apache, MySQL y PHP](#)

[Activación de la seguridad en Apache](#)

[Plugin PHPclipse en Debian](#)

[MySql en Windows](#)

Descripción

Alejandro Perez nos enseña como potenciar el entorno eclipse para facilitarnos el trabajo con ficheros xml, gracias al pluggin X-MEN

Os mostamos como instalar esta potente herramienta gratuita para la creación de imágenes y el retoque de imágenes.

Os mostramos como configurar Apache, MySQL y PHP en vuestra máquina

Alejandro Pérez nos enseña como securizar Apache a través de autenticación básica y certificados de seguridad SSL.

Alejandro Pérez nos muestra como instalar el plugin de Eclipse: PHPclipse, en GNU / Linux (Debian)

MySql es una de las principales bases de datos "gratuitas" que podemos encontrar en Internet. En este tutorial aprenderéis a instalarlo en Windows

Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)



www.AdictosAlTrabajo.com Optimizado 800X600