

Creación de tablas

Las tablas son *elementos* de las base de datos. Por esa razón *nos resultará imposible crear* una **tabla** sin tener *creada y seleccionada* una *base de datos*.

Es por eso que para la creación de una tabla se necesitan los siguientes requisitos:

- Tener abierta una conexión con el servidor MySQL.

- Tener seleccionada una base de datos.

Conexión con el servidor

La hemos comentado en la página anterior, recuerda que requería la función:

```
$c=mysql_connect(h, u, p)
```

Esa conexión ha de ser establecida antes de cualquier otra intervención relacionada con accesos a bases de datos y tablas.

Selección de la base de datos

Dado que podemos manejar bases de datos distintas –en este curso usaremos *ejemplos* y *practicar*– es preciso **decir** a MySQL con qué base queremos trabajar.

```
mysql_select_db("n", $c);
```

donde **n** es el nombre de la base de datos (puede ser una cadena entrecomillada o el nombre de una variable previa que contenga ese nombre). En este último caso, como es habitual, el nombre de la variable no llevaría comillas.

El segundo parámetro **\$c** es el *identificador de conexión*. Es decir, la variable creada al establecer la conexión con MySQL.

Este valor debe insertarse siempre. La razón es que **MySQL** permite mantener abiertas, de forma simultánea, varias conexiones (podríamos manejar más de un servidor de bases de datos) y en esas condiciones sería necesaria una *conexión distinta* para cada servidor.

Creación de una tabla

En todas las transacciones **PHP-MySQL** habremos de utilizar instrucciones de ambos lenguajes.

Crear una tabla MySQL

La creación de tablas MySQL requiere una de estas dos sentencias:

```
CREATE TABLE IF NOT EXISTS tabla (campo1, campo2,... )
```

```
CREATE TABLE tabla (campo1, campo2,... )
```

La única diferencia entre ambas opciones es que la segunda daría un error si tratáramos de crear una tabla preexistente (deberíamos recurrir al procedimiento que hemos visto cuando creábamos bases de datos) mientras que la primera no da ese mensaje de error.

Aunque no lo hemos indicado, **CREATE DATABASE** también permite esta sintaxis alternativa.

Definición de campos en una tabla MySQL

Cada uno de los campos que vayamos a crear en una tabla requiere una definición que debe contener lo siguiente:

– nombre del campo

Es una palabra cualquiera –distinta para campo de la tabla y que normalmente suele elegirse aludiendo al contenido. Por ejemplo, *fec_nac*, *nom_perro*, etcétera.

No va entre comillas nunca y MySQL **diferencia** mayúsculas/minúsculas.

Para utilizar como nombres de campo **palabras reservadas** –por ejemplo, **create**– del lenguaje MySQL debemos escribirla entre ` `. Observa que no son comillas sino *acentos graves*. Lo más aconsejable es evitar esta situación.

– tipo y dimensiones

Los tipos de campos –los hemos visto en la *página anterior*– tienen que ajustarse a uno de los soportados por MySQL. Se escriben a continuación del nombre **sin otra separación que un espacio** y requieren la sintaxis –estricta– correspondiente a cada tipo.

– **flags del campo (son opcionales)** Puede utilizarse cualquiera de los permitidos para cada tipo de campo –*puedes verlos* encerrados entre corchetes– al lado de cada tipo de campo.

Ejemplo de creación de una tabla

```
<?
/* nos conectamos con el servidor
recogiendo en $c el identificador de conexión */
$c=mysql_connect ("localhost","pepe","pepa") or die ("Imposible conectar")
# seleccionamos una base de datos existente
# de lo contrario nos daría un error
# pondremos como nombre ejemplos nuestra base de datos
# creada en la página anterior y usaremos $c
# importante no olvidarlo
mysql_select_db ("ejemplos", $c);
/* ahora ya estamos en condiciones de crear la tabla
podríamos escribir ya la instrucción mysql_query y meter
detró la sentencia MySQL pero, por razones de comodidad
crearemos antes una variable que recoja toda la sentencia
y será luego cuando la ejecutemos.
Definiremos una variable llamada $crear e iremos añadiendo cosas */
# la primera parte de la instrucción es esta (espacio final incluido)
$crear="CREATE TABLE IF NOT EXISTS ";
# añadiremos el nombre de la tabla que será ejemplo1
# fijate en el punto (concatenador de cadenas) que permite
# ir añadiendo a la cadena anterior
$crear.="ejemplo1 ";
```

La forma habitual –hay algunas excepciones– en la que **PHP** se comunica con **MySQL** es la función:

```
mysql_query("sent", $c);
```

donde la cadena **sent** contiene las instrucciones propias de MySQL –las comentamos al margen– y **\$c** sigue siendo la variable que contiene el identificador de conexión.

La sintaxis de las sentencias MySQL que crean tablas la tenemos en la columna de la derecha.

DIMENSIONES

Cuando se establezca una **dimensión** como parámetro opcional de un campo deben tenerse en cuenta algunos detalles.

Si en un campo numérico introdujéramos valores que exceden los límites su valor no se registraría en la tabla sino el valor del *límite más próximo* correspondiente a ese tipo de campo.

Supongamos un campo tipo TINYINT que permite valores comprendidos entre -128 y 127. Si asignáramos en uno de sus registros un valor igual a **234** se escribiría en la tabla **127** (el límite superior) y si ponemos **-834** escribiría el límite inferior, es decir, -128.

En caso de cadenas, si el valor introducido sobrepasara la longitud máxima permitida, la cadena sería *recortada* y únicamente se registraría el número máximo de caracteres permitidos.

Consideraciones sobre IF NOT EXISTS

Esta variante de CREATE –aplicable tanto en tablas como en bases de datos– tiene la ventaja de no dar mensajes de error en caso de intentar crear una tabla –o base– ya existente, pero puede darnos algún sobresalto porque **no advierte** que la tabla no ha sido creada y puede darnos la sensación de que puede haber reescrito una tabla anterior.

Si pulsas reiteradamente en el enlace *Crear tabla ejemplo1* comprobarás que *no aparece ningún mensaje de error*.

Crear la TABLA
EJEMPLO1

```
#ahora pongamos el paréntesis (con un espacio delante)
#aunque el espacio también podría detrás de ejemplo1
$crear .="( ";
# insertemos el primer campo y llamémoslo num1
# hagámoslo de tipo TINYINT sin otras especificamos
# sabiendo que solo permitira valores numéricos
# comprendidos entre -128 y 127
$crear .="num1 TINYINT , ";
# LOS CAMPOS SE SEPARAN CON COMAS por eso
# la hemos incluido al final de la instrucción anterior

# ahora num2 del mismo tipo con dimensión 3 y el flag UNSIGNED
# Y ZEROFILL que: cambiará los límites de valores
# al intervalo 0 - 255, y rellenará con ceros por la izquierda
# en el caso de que el número de cifras significativas
# sea menor de 3.
# Fíjate que los flags van separado únicamente por espacios
$crear .="num2 TINYINT (3) UNSIGNED ZEROFILL, ";
# en num3 identico al anterior añadiremos un valor por defecto
# de manera que cuando se añadan registros a la tabla
# se escriba automaticamente ese valor 13 en el caso
# de que no le asignemos ninguno a ese campo
# por ser numérico 13 no va entre comillas
$crear .="num3 TINYINT (7) UNSIGNED ZEROFILL DEFAULT 13, ";
# ahora un número decimal num4 tipo REAL con 8 dígitos en total
# de los cuales tres serán decimales y también rellenaremos con ceros
# Pondremos como valor por defecto 3.14
$crear .="num4 REAL (8,3) ZEROFILL DEFAULT 3.14, ";
# añadamos una fecha
$crear .="fecha DATE, ";
/* una cadena con un límite de 32 carácter con BINARY
para que diferencie Pepe de PEPE */
$crear .="cadena VARCHAR(32) BINARY, ";
/* un último campo -opcion- del tipo ENUM que solo admita
como valores SI, NO, QUIZA
fíjate en las comillas y en el parentesis
¡¡cuidado...!! aquí no ponemos coma al final
es el último campo que vamos a insertar y no necesita
ser separado. Si la pones dará un ERROR */
$crear .="opcion ENUM('Si','No','Quiza') ";
# solo nos falta añadir el paréntesis conteniendo toda la instrucción
$crear .=")";
/* tenemos completa la sentencia MySQL
solo falta ejecutarla mediante mysql_query
ya que la conexión está abierta
y la base de datos ya está seleccionada */

/* pongamos un condicional de comprobación */
if(mysql_query($crear,$c)){
    print "Se ha creado la base de datos<br>";
    print "La sentencia MySQL podríamos haberla escrito asi:<br>";
    print "mysql_query(\".\"\".\".$crear.\"\" , $c);";
}else{
    print "Se ha producido un error al crear la tabla";
}
?>
```

Ejercicio nº 38

Como práctica de creación de tablas, deberás crear –en tu base de datos **practicas**– una tabla a la que llamaremos **tabla1** que recoja –al menos– en diferentes campos, los siguientes datos de tus alumnos: DNI (con letra incluida), nombre, apellidos (en campos diferentes), fecha de nacimiento, y –en un campo tipo ENUM– su condición de repetidor o no.

Anterior

Índice

Siguiente