

Lenguaje PHP

DESARROLLO BACK-END

Acceso a Base de Datos - MySQL

1. Acceso a una base de datos MySQL a través de

Acceso a bases de datos MySQL en PHP

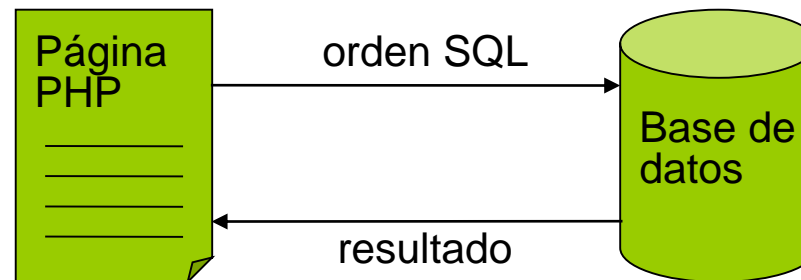
1. Herramientas de administración: phpMyAdmin
2. Lenguaje SQL
3. Funciones de PHP para el acceso a bases de datos MySQL

Acceso a bases de datos MySQL en PHP

- **phpMyAdmin** es una herramienta para la administración del servidor de bases de datos MySQL
- Dispone de una interfaz gráfica y es de libre distribución
- Permite realizar todo tipo de operaciones sobre bases de datos:
 - crear, borrar y modificar tablas
 - consultar, insertar, modificar y eliminar datos
 - definir usuarios y asignar permisos
 - realizar copias de seguridad
 - etc
- Está escrita en php y se ejecuta desde el navegador
- Si está instalada en la carpeta phpmyadmin, se ejecuta escribiendo en la barra de direcciones del navegador la url
`http://localhost/phpmyadmin/`
- Puede administrar bases de datos locales y remotas

Acceso a bases de datos MySQL en PHP

- SQL (*Structured Query Language*) es el lenguaje que se utiliza para comunicarse con la base de datos
- Procedimiento de comunicación con la base de datos:



Acceso a bases de datos MySQL en PHP

- Los pasos para acceder desde PHP a una base de datos son los siguientes:
 - Conectar con el servidor de bases de datos
 - Seleccionar una base de datos
 - Enviar la instrucción SQL a la base de datos
 - Obtener y procesar los resultados
 - Cerrar la conexión con el servidor de bases de datos

Acceso a bases de datos MySQL en PHP

- Existen cuatro maneras de acceder a una base de datos MySQL:
- Utilizando las funciones `mysql`. Este metodo esta definido como obsoleto a partir de la version 5.5.0 de PHP y fue removido a partir de la version 7.0.0 por lo tanto no es recomendable su utilizacion, pero podemos encontrarnos con mucho codigo que aun hace uso de ellas.
- Utilizando las funciones `mysqli`. Son el reemplazo natural de las funciones `mysql`, se pueden utilizar de dos maneras:
 - Formato procedural
 - Orientado a Objetos
- Utilizando PDO (PHP Data Objects). La libreria de acceso a datos PDO nos entrega una capa de abstraccion de datos que nos permite utilizar el mismo codigo de acceso a datos sin importar el motor de base de datos que utilicemos

Acceso a bases de datos MySQL en PHP

MySQL Procedural

Conectar con el servidor de bases de datos y seleccion de base de datos:

mysqli_connect()

- Devuelve un objeto de la conexión en caso de éxito y false en caso contrario

- Sintaxis:

```
$conexion = mysqli_connect(servidor, usuario, password, bd );
```

- Ejemplo:

```
$conexion = mysqli_connect("localhost", "cursophp", "", "base")  
or die ("No se puede conectar con el servidor");
```


Acceso a bases de datos MySQL en PHP

MySQL Procedural

Enviar la instrucción SQL a la base de datos: *mysqli_query(conexion, query)*

- Devuelve un identificador o true (dependiendo de la instrucción) si la instrucción se ejecuta correctamente y false en caso contrario
- Sintaxis:

```
$consulta = mysqli_query(conexion, instruccion sql);
```

Ejemplo:

```
$conexion = mysqli_connect("localhost", "cursophp", "", "base")  
or die ("No se puede conectar con el servidor");  
$consulta = mysqli_query($conexion, "select * from noticias") or  
die ("Fallo en la consulta");
```

Acceso a bases de datos MySQL en PHP

- **Obtener y procesar los resultados:** `mysqli_num_rows()`, `mysqli_fetch_assoc()`
- En el caso de que la instrucción enviada produzca unos resultados, `mysqli_query()` devuelve las filas de la tabla afectadas por la instrucción
 - `mysqli_num_rows()` devuelve el número de filas afectadas
 - Para obtener las distintas filas del resultado se utilizan la función `mysqli_fetch_assoc()`

- **Sintaxis:**

```
$nfilas = mysqli_num_rows(linkConsulta);  
$fila = mysqli_fetch_assoc(linkConsulta);
```

Ejemplo:

```
$conexion = mysqli_connect("localhost", "cursophp", "", "base")  
$consulta = mysqli_query($conexion, "select * from noticias")  
$nfilas = mysqli_num_rows($consulta);  
$fila = mysqli_fetch_assoc($consulta);
```

Acceso a bases de datos MySQL en PHP

- Ejemplo:

noticias				
Id	titulo	texto	tipo	fecha
1	Título 2	Texto 2	promociones	05/02/2004
2	Título 3	Texto 3	promociones	04/02/2004
3	Título 4	Texto 4	costas	01/02/2004
5	Título 5	Texto 5	promociones	31/01/2004

Instrucción:

```
select * from noticias where categoria="promociones"
```

Acceso a bases de datos MySQL en PHP

- Ejemplo:

noticias				
Id	titulo	texto	tipo	fecha
1	Título 2	Texto 2	promociones	05/02/2004
2	Título 3	Texto 3	promociones	04/02/2004
3	Título 4	Texto 4	costas	01/02/2004
5	Título 5	Texto 5	promociones	31/01/2004



\$consulta
\$nfilas=3

Instrucción:

```
select * from noticias where categoria="promociones"
```

Acceso a bases de datos MySQL en PHP

```
$conexion = mysqli_connect("localhost", "root", "", "diario");  
$sql = "select * from noticias where categoria='promociones' ";  
$consulta = mysqli_query($conexion, $sql);  
$nfilas = mysqli_num_rows($consulta);  
echo $nfilas;  
while($fila = mysqli_fetch_assoc($consulta)) {  
    echo $fila["Id"];  
    echo $fila["texto"];  
}
```

Acceso a bases de datos MySQL en PHP

- Cerrar la conexión con el servidor de bases de datos:
`mysqli_close()`

- Sintaxis:

```
mysqli_close($conexion);
```

- Ejemplo

```
$conexion = mysqli_connect("localhost", "root", "",  
    "diario");  
$sql = "select * from noticias where categoria='promociones' ";  
$consulta = mysqli_query($conexion, $sql);  
$nfilas = mysqli_num_rows($consulta);  
echo $nfilas . "<br>";  
while($fila = mysqli_fetch_assoc($consulta)) {  
    echo $fila["Id"];  
    echo $fila["texto"] . "<br>";  
}  
mysqli_close($conexion);
```

Acceso a bases de datos MySQL en PHP

MySQL Orientado a Objetos – Conexión

```
<?php
$server = "localhost";
$user = "usuario";
$password = "clave";

$conn = new mysqli($server, $user, $password);

if ($conn->connect_error) {
    die("Fallo la conexión: " . $conn->connect_error);
}
echo "Conexión exitosa";
?>
```

Acceso a bases de datos MySQL en PHP

MySQL Orientado a Objetos – Inserción de Datos

```
<?php
    ... Conexión

$sql = "INSERT INTO tabla (campo1, campo2, campo3)
VALUES ('Juan', 'Perez', 'juan@example.com')";

if ($conn->query($sql) === TRUE) {
    echo "Registro Creado";
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}

$conn->close();
?>
```


Acceso a bases de datos MySQL en PHP

MySQL Orientado a Objetos – Obtención de el ultimo ID

```
<?php
```

```
... Conexión
```

```
$sql = "INSERT INTO tabla (campo1, campo2, campo3)  
VALUES ('Juan', 'Perez', 'juan@example.com')";
```

```
if ($conn->query($sql) === TRUE) {  
    $last_id = $conn->insert_id;  
    echo "La ultima Id Insertada: " . $last_id;  
} else {  
    echo "Error: " . $sql . "<br>" . $conn->error;  
}
```

```
$conn->close();  
?>
```

Acceso a bases de datos MySQL en PHP

MySQL Orientado a Objetos – Obtener Registros

```
<?php
```

```
... Conexión
```

```
$sql = "SELECT id, nombre, apellido FROM tabla";  
$result = $conn->query($sql);  
  
if ($result->num_rows > 0) {  
    while($row = $result->fetch_assoc()) {  
        echo "id: " . $row["id"] . " - Nombre: " .  
$row["nombre"] . " " . $row["apellido"] . "<br>";  
    }  
} else {  
    echo "Resultado: 0";  
}  
$conn->close();  
?>
```

Acceso a bases de datos MySQL en PHP

MySQL Orientado a Objetos – Prepared Statments

```
<?php
```

```
... Conexión
```

```
$stmt = $conn->prepare("INSERT INTO tabla (nombre,  
apellido, email) VALUES (?, ?, ?)");  
$stmt->bind_param("sss", $nombre, $apellido, $email);
```

```
$nombre = "Juan";  
$apellido = "Perez";  
$email = "juan@example.com";  
$stmt->execute();  
  
$nombre = "Maria";  
$apellido = "García";  
$email = "maria@example.com";  
$stmt->execute();
```

```
$stmt->close();  
$conn->close();  
?>
```