

Cómo utilizar un wrapper o envoltorio de la clase mysqli

Un envoltorio (wrapper) mysqli

Introducción

A veces cuando utilizamos la clase mysqli se hace bastante complicado el escribir todas las instrucciones para hacer una simple consulta. Es por ello que existen numeros wrappers o envoltorios, que nos permiten utilizar la clase mysqli, de una forma mucho más sencilla.

En la dirección <http://www.phpclasses.org/> podremos encontrar un montón de wrappers para usar con mysqli. Si buscamos por mysqli wrapper: http://www.phpclasses.org/search.html?words=mysqli+wrapper&x=0&y=0&go_search=1 podremos descargar gratuitamente (previo registro), el código fuente de dichas clases para utilizar en nuestros proyectos.

Ejemplo de wrapper mysqli

Aquí se muestra un ejemplo de utilización de un wrapper mysqli.

Para ello descargamos primero el código fuente de la clase, desde aquí: [Archivo:MySqlIwrapper.zip](#)

El archivo mysqliwrapper.zip ha sido obtenido de: <https://github.com/carldanley/PHP-MySQLi-Wrapper> dónde podremos encontrar más información de uso y utilización de la clase.

Esta clase utiliza un fichero de configuración como el utilizado en clases para el proyecto Amadeus.

El siguiente fichero se puede adaptar fácilmente a nuestra clase de configuración, simplemente modificando en nuestra clase el nombre de las variables utilizadas para conectarnos a la base de datos.

Contenido del fichero **class-config-example.php**.

```
<?php
/**
 * Original Filename: class-config-example.php
 * User: carldanley
 * Created on: 1/21/13
 * Time: 2:03 PM
 */

class Config {
public static $db_host = 'localhost';
public static $db_username = 'usuariodb';
public static $db_password = 'xxxxxxx';
public static $db_database = 'basedatos';
}
?>
```

Ejemplo de uso de la clase **Database (wrapper mysqli)**:

```
<?php
require_once 'class-config-example.php';
require_once 'class-database.php';

echo "<h2>Uso de un wrapper mysqli.</h2>";

// Más información en: https://github.com/carldanley/PHP-MySQLi-Wrapper

// == Obtener datos de una tabla ==
// Conectamos a la base de datos:
Database::connect();

// Condición de búsqueda
Database::where( 'nombre', 'Rafa Veiga' );

// Obtenemos un array con todos los registros.
```

```

$registros = Database::get( 'amadeus_usuarios' );

// Imprimimos el contenido del array.
print_r($registros);

// Imprimimos el nick del primer registro.
echo "<br/>Nick: ".$registros[0]['nick']."<br/>";

// Cerramos la conexión con la base de datos.
Database::disconnect();

// == Insertar datos en una tabla ==
// Conectamos a la base de datos:
Database::connect();

// Datos a insertar
$datos=array('nick'=>'martita','nombre'=>'Marta','apellidos'=>'Suarez Diaz');

// Insertamos un array con todos los registros.
Database::insert('amadeus_usuarios',$datos);
echo "<br/>Datos insertados correctamente";

// Cerramos la conexión con la base de datos.
Database::disconnect();

// == Borrar datos de una tabla ==
// Conectamos a la base de datos:
Database::connect();

// Condición de borrado
Database::where( 'nick','trolebus' );

// Borramos todos los registros
Database::delete('amadeus_usuarios');
echo "<br/>Datos borrados correctamente";

// Cerramos la conexión con la base de datos.
Database::disconnect();

// == Consultas avanzadas ==
// Conectamos a la base de datos:
Database::connect();

// Obtenemos un array con todos los registros.
$registros = Database::query( "select * from amadeus_usuarios where nombre like 'r%' ");

// Imprimimos todos los registros.
foreach($registros as $dato)
{
    echo "<br/><br/>Nick: ".$dato['nick'];
    echo "<br/>Nombre: ".$dato['nombre'];
    echo "<br/>Apellidos: ".$dato['apellidos'];
    echo "<br/>E-mail: ".$dato['email'];
}

// Cerramos la conexión con la base de datos.
Database::disconnect();
?>

```

--Veiga (discusión) 01:50 15 oct 2014 (CEST)