

La Extensión Mejorada de MySQL (MySQLi) proporciona una Interfaz Procedimental así como una Interfaz Orientada a Objetos. En este artículo, vamos a explorar algunas de las funciones procedimentales comunes de MySQLi.

NOTA IMPORTANTE: *Para que el tratamiento de errores funcione al modo tradicional y no genere una excepción por defecto, a partir de la versión de Php 8.2 hay que incluir esta línea al principio del Script*

```
mysqli_report(MYSQLI_REPORT_OFF); //Para desabilitar las excepciones en versiones posteriores a 8.1
```

mysqli_connect() Esta función se utiliza para conectarse a MySQL. Antes de realizar cualquier operación en la base de datos, necesitas conectarte a MySQL. En caso de éxito, esta función devuelve un identificador de enlace que puedes usar en otras funciones de MySQLi. En caso de fallo, lanzará un error.

Aquí está cómo un usuario llamado robin con la contraseña robin123 necesita conectarse a una base de datos llamada company_db en localhost. El usuario robin debe tener privilegios para acceder a company_db.

phpCopy code

```
$link = mysqli_connect('localhost', 'robin', 'robin123', 'company_db');
```

Si tu puerto MySQL es diferente al predeterminado (3308), necesitas proporcionar el número de puerto como quinto parámetro.

phpCopy code

```
$link = mysqli_connect('localhost', 'robin', 'robin123', 'company_db', '3800');
```

mysqli_connect_error() mysqli_connect() arroja un error en caso de fallo y mysqli_connect_error() almacena el error en la última llamada a mysqli_connect(). Si no hay error, devuelve NULL.

Para probar un error de mysqli_connect(), detén el servidor MySQL y llama a mysqli_connect(). Si has habilitado los errores de PHP, verás un error que incluye información como la siguiente.

mysqli_connect_error() devolvería el mismo mensaje.

vbnetCopy code

```
Can't connect to MySQL server on 'localhost'
```

En la práctica, no es recomendable mostrar mensajes de error como estos a los usuarios de tu aplicación PHP (pueden contener datos sensibles y parecer muy técnicos). Por lo tanto, puedes usar el operador de suspensión de errores delante de mysqli_connect() para evitar que arroje errores y usar mysqli_connect_error() para registrar el error para la solución de problemas, como se muestra a continuación:

phpCopy code

```
$link = @mysqli_connect('localhost', 'robin', 'robin123', 'company_db');
```

```

if (mysqli_connect_error()) {
    $logMessage = 'Error de MySQL: ' . mysqli_connect_error();
    // Llama a tu registrador aquí.
    die('No se pudo conectar a la base de datos');
}

```

```
// El resto del código va aquí.
```

mysqli_select_db() Para cambiar la base de datos en uso, puedes usar `mysqli_select_db()`. Por ejemplo, supongamos que el usuario robin también tiene privilegios para una base de datos llamada `company_new_db`, entonces puedes cambiar la base de datos de la siguiente manera:

phpCopy code

```
$link = @mysqli_connect('localhost', 'robin', 'robin123', 'company_db');
```

```
// Operaciones en 'company_db'
```

```
mysqli_select_db($link, 'company_new_db');
```

```
// Operaciones en 'company_new_db'
```

Solo necesitarás esta función si tu aplicación PHP trabaja con más de una base de datos

mysqli_close() Puedes usar esta función para cerrar una conexión de MySQL. Devuelve TRUE en caso de éxito y FALSE en caso de fallo.

phpCopy code

```
$link = @mysqli_connect('localhost', 'robin', 'robin123', 'company_db');
```

```
// Operaciones de MySQL van aquí.
```

```
mysqli_close($link);
```

PHP cerrará las conexiones abiertas y liberará los recursos al finalizar tu script de PHP. Sin embargo, es una buena práctica usar explícitamente `mysqli_close()` al finalizar las operaciones de MySQL para liberar los recursos de inmediato.

mysqli_query() Esta es la función utilizada para ejecutar consultas de MySQL. Devuelve FALSE en caso de fallo. Para consultas SELECT, SHOW, DESCRIBE y EXPLAIN (donde hay una salida), devuelve un conjunto de resultados de MySQL (recurso) que se puede usar en funciones como `mysqli_fetch_array()`. Para todas las demás consultas como INSERT, UPDATE y DELETE, devuelve TRUE en caso de éxito.

phpCopy code

```
$link = @mysqli_connect('localhost', 'robin', 'robin123', 'company_db');
```

```
$query = "SELECT * FROM employee";
```

```

if (mysqli_query($link, $query)) {
    // Iterar y mostrar el resultado.
} else {
    // Mostrar error.
}

```

```
mysqli_close($link);
```

mysqli_fetch_array() Esta función se utiliza para leer datos de un conjunto de resultados de MySQL (devuelto por un `mysqli_query()`). Lee y devuelve una fila de datos como un array y luego mueve el puntero a la siguiente fila. Cuando no hay más filas que devolver, devuelve NULL. Debido a este comportamiento, a menudo se usa con un bucle While como se muestra a continuación.

phpCopy code

```
while ($row = mysqli_fetch_array($result)) {  
    /* Mientras haya datos, $row será un array.  
    Al final, $row se convierte en NULL terminando el bucle. */  
}
```

Supongamos que la siguiente tabla de empleados está disponible en nuestra base de datos `company_db`.

Aquí está cómo obtendríamos el ID, el Primer Nombre y el Apellido de esta tabla.

phpCopy code

```
$link = @mysqli_connect('localhost', 'robin', 'robin123', 'company_db');  
$query = "SELECT `id`, `first_name`, `last_name` FROM `employee`";  
$result = mysqli_query($link, $query);
```

```
while ($row = mysqli_fetch_array($result)) {  
    echo $row[0] . ': ' . $row[1] . ' ' . $row[2];  
    echo '<br />';  
}
```

```
mysqli_free_result($result);  
mysqli_close($link);
```

Cuando ejecutes el código anterior, se mostrará el siguiente contenido en el navegador web.

makefileCopy code

```
1: Robin Jackman  
2: Taylor Edward
```

Además de un array indexado, `mysqli_fetch_array()` también devuelve un array asociativo donde las claves son los nombres de columna correspondientes de la tabla. Por lo tanto, el siguiente segmento de código producirá el mismo resultado.

phpCopy code

```
while ($row = mysqli_fetch_array($result)) {  
    echo $row['id'] . ': ' . $row['first_name'] . ' ' . $row['last_name'];  
    echo '<br />';  
}
```

Puedes limitar qué array devolver como se muestra a continuación.

phpCopy code

```
// Devuelve solo un array indexado
mysqli_fetch_array($result, MYSQLI_NUM);
```

```
// Devuelve solo un array asociativo
mysqli_fetch_array($result, MYSQLI_ASSOC);
```

PHP proporciona dos funciones que producen los mismos resultados que obtienes al pasar constantes a `mysqli_fetch_array()`.

```
phpCopy code
// Lo mismo que mysqli_fetch_array($result, MYSQLI_NUM)
mysqli_fetch_row($result);

// Lo mismo que mysqli_fetch_array($result, MYSQLI_ASSOC)
mysqli_fetch_assoc($result);
```

mysqli_free_result() Inmediatamente después de usar un conjunto de resultados, puedes liberar la memoria utilizada para ello de la siguiente manera.

```
phpCopy code
mysqli_free_result($result);
```

mysqli_num_rows() `mysqli_num_rows()` devuelve el número de filas en un conjunto de resultados. Usándolo, puedes tomar una acción diferente cuando el conjunto de resultados está vacío.

```
phpCopy code
if (mysqli_num_rows($result) > 0) {
    // Continuar con el $result
} else {
    // Mostrar un mensaje de error
}
```

mysqli_affected_rows() Esta función proporciona información sobre la última consulta de MySQL ejecutada. Para INSERT, UPDATE, REPLACE y DELETE, proporciona el número de filas afectadas. Para SELECT, devuelve el número de filas en el conjunto de resultados como `mysqli_num_rows()`.

Por ejemplo, la siguiente es una consulta de UPDATE para actualizar el apellido de Taylor en la tabla de empleados. Siendo el campo id único, sabemos que solo se debería afectar una fila con esta consulta.

```
phpCopy code
$query = "UPDATE `employee` SET `last_name` = 'Adams' WHERE `id` = 2";
mysqli_query($link, $query);

if (mysqli_affected_rows($link) == 1) {
    // Resto del código
} else {
    // Mostrar un mensaje de error
}
```

mysqli_error() Si hubo un error en la última consulta de MySQL, esta función devolverá el error. Si no hubo error, devolverá una cadena vacía.

phpCopy code

```
if (!mysqli_query($link, $query)) {  
    $logMessage = 'Error de MySQL: ' . mysqli_error($link);  
    // Llama a tu registrador aquí.  
    die('Hubo un error en la consulta');  
}
```