

**mSQL**

# Introduction

These functions allow you to access mSQL database servers. More information about mSQL can be found at [» http://www.hughes.com.au/](http://www.hughes.com.au/).

# Installing/Configuring

## Requirements

No external libraries are needed to build this extension.

## Installation

In order to have these functions available, you must compile PHP with msql support by using the `--with-msql[=DIR]` option. DIR is the mSQL base install directory, defaults to `/usr/local/msql3`.

### Note

#### Note to Win32 Users

In order for this extension to work, there are DLL files that must be available to the Windows system *PATH*. For information on how to do this, see the FAQ entitled "[How do I add my PHP directory to the PATH on Windows](#)". Although copying DLL files from the PHP folder into the Windows system directory also works (because the system directory is by default in the system's *PATH*), this is not recommended. *This extension requires the following files to be in the PATH: msql.dll*

## Runtime Configuration

The behaviour of these functions is affected by settings in *php.ini*.

### mSQL configuration options

Name	Default	Changeable	Changelog
msql.allow_persistent	"1"	PHP_INI_ALL	
msql.max_persistent	"-1"	PHP_INI_ALL	
msql.max_links	"-1"	PHP_INI_ALL	

For further details and definitions of the `PHP_INI_*` constants, see the [php.ini directives](#).

Here's a short explanation of the configuration directives.

*mysql.allow\_persistent* **boolean**

Whether to allow persistent mSQL connections.

*mysql.max\_persistent* **integer**

The maximum number of persistent mSQL connections per process.

*mysql.max\_links* **integer**

The maximum number of mSQL connections per process, including persistent connections.

## Resource Types

There are two resource types used in the mSQL module. The first one is the link identifier for a database connection, the second a resource which holds the result of a query.

# Predefined Constants

The constants below are defined by this extension, and will only be available when the extension has either been compiled into PHP or dynamically loaded at runtime.

**MSQL\_ASSOC** ( [integer](#) )

**MSQL\_NUM** ( [integer](#) )

**MSQL\_BOTH** ( [integer](#) )

# Examples

This simple example shows how to connect, execute a query, print resulting rows and disconnect from a mSQL database.

## Example #1 - mSQL usage example

```
<?php
/* Connecting, selecting database */
$link = msql_connect('localhost', 'username', 'password')
    or die('Could not connect : ' . msql_error($link));

msql_select_db('database', $link)
    or die('Could not select database');

/* Issue SQL query */
$query = 'SELECT * FROM my_table';
$result = msql_query($query, $link) or die('Query failed : ' .
    msql_error());

/* Printing results in HTML */
echo "<table>\n";
while ($row = msql_fetch_array($result, MSQL_ASSOC)) {
    echo "\t<tr>\n";
    foreach ($row as $col_value) {
        echo "\t\t<td>$col_value</td>\n";
    }
    echo "\t</tr>\n";
}
echo "</table>\n";

/* Free result set */
msql_free_result($result);

/* Close connection */
msql_close($link);
?>
```

# mSQL Functions

# mysql\_affected\_rows

mysql\_affected\_rows -- Returns number of affected rows

## Description

int **mysql\_affected\_rows** ( resource \$result )

Returns number of affected rows by the last SELECT, UPDATE or DELETE query associated with *result*.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

## Return Values

Returns the number of affected rows on success, or **FALSE** on error.

## See Also

- [mysql\\_query\(\)](#)



# mysql\_close

mysql\_close -- Close mSQL connection

## Description

bool **mysql\_close** ( [ resource \$link\_identifier ] )

[mysql\\_close\(\)](#) closes the non-persistent connection to the mSQL server that's associated with the specified link identifier.

Using [mysql\\_close\(\)](#) isn't usually necessary, as non-persistent open links are automatically closed at the end of the script's execution. See also [freeing resources](#).

## Parameters

*link\_identifier*

The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns **TRUE** on success or **FALSE** on failure.

## See Also

- [mysql\\_connect\(\)](#)
- [mysql\\_pconnect\(\)](#)

# mysql\_connect

mysql\_connect -- Open mSQL connection

## Description

resource **mysql\_connect** ( [ string \$hostname ] )

[mysql\\_connect\(\)](#) establishes a connection to a mSQL server.

In case a second call is made to [mysql\\_connect\(\)](#) with the same arguments, no new link will be established, but instead, the link identifier of the already opened link will be returned.

The link to the server will be closed as soon as the execution of the script ends, unless it's closed earlier by explicitly calling [mysql\\_close\(\)](#).

## Parameters

*hostname*

The hostname can also include a port number. e.g. *hostname,port*. If not specified, the connection is established by the means of a Unix domain socket, being then more efficient than a localhost TCP socket connection.

Note
While this function will accept a colon (:) as a host/port separator, a comma (,) is the preferred method.

## Return Values

Returns a positive mSQL link identifier on success, or **FALSE** on error.

## See Also

- [mysql\\_pconnect\(\)](#)
- [mysql\\_close\(\)](#)

# mysql\_create\_db

mysql\_create\_db -- Create mSQL database

## Description

bool **mysql\_create\_db** ( string \$database\_name [, resource \$link\_identifier ] )

[mysql\\_create\\_db\(\)](#) attempts to create a new database on the mSQL server.

## Parameters

*database\_name*

The name of the mSQL database.

*link\_identifier*

The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns **TRUE** on success or **FALSE** on failure.

## See Also

- [mysql\\_drop\\_db\(\)](#)

# mysql\_createdb

mysql\_createdb -- Alias of [mysql\\_create\\_db\(\)](#)

## Description

This function is an alias of: [mysql\\_create\\_db\(\)](#).

# mysql\_data\_seek

mysql\_data\_seek -- Move internal row pointer

## Description

bool **mysql\_data\_seek** ( resource \$result, int \$row\_number )

[mysql\\_data\\_seek\(\)](#) moves the internal row pointer of the mSQL result associated with the specified query identifier to point to the specified row number. The next call to [mysql\\_fetch\\_row\(\)](#) would return that row.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*row\_number*

The seeked row number.

## Return Values

Returns **TRUE** on success or **FALSE** on failure.

## See Also

- [mysql\\_fetch\\_array\(\)](#)
- [mysql\\_fetch\\_object\(\)](#)
- [mysql\\_fetch\\_row\(\)](#)
- [mysql\\_result\(\)](#)

# mysql\_db\_query

mysql\_db\_query -- Send mSQL query

## Description

resource **mysql\_db\_query** ( string \$database, string \$query [, resource \$link\_identifier ] )

[mysql\\_db\\_query\(\)](#) selects a database and executes a query on it.

## Parameters

*database*

The name of the mSQL database.

*query*

The SQL query.

*link\_identifier*

The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns a positive mSQL query identifier to the query result, or **FALSE** on error.

## See Also

- [mysql\\_query\(\)](#)

# mysql\_dbname

mysql\_dbname -- Alias of [mysql\\_result\(\)](#)

## Description

This function is an alias of: [mysql\\_result\(\)](#).

# mysql\_drop\_db

mysql\_drop\_db -- Drop (delete) mSQL database

## Description

bool **mysql\_drop\_db** ( string \$database\_name [, resource \$link\_identifier ] )

[mysql\\_drop\\_db\(\)](#) attempts to drop (remove) a database from the mSQL server.

## Parameters

*database\_name*

The name of the database.

*link\_identifier*

The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns **TRUE** on success or **FALSE** on failure.

## See Also

- [mysql\\_create\\_db\(\)](#)



# mysql\_error

mysql\_error -- Returns error message of last mysql call

## Description

string **mysql\_error** ( void )

[mysql\\_error\(\)](#) returns the last issued error by the mSQL server. Note that only the last error message is accessible with [mysql\\_error\(\)](#).

## Return Values

The last error message or an empty string if no error was issued.

# mysql\_fetch\_array

mysql\_fetch\_array -- Fetch row as array

## Description

array **mysql\_fetch\_array** ( resource \$result [, int \$result\_type ] )

[mysql\\_fetch\\_array\(\)](#) is an extended version of [mysql\\_fetch\\_row\(\)](#). In addition to storing the data in the numeric indices of the result array, it also stores the data in associative indices, using the field names as keys.

An important thing to note is that using [mysql\\_fetch\\_array\(\)](#) is NOT significantly slower than using [mysql\\_fetch\\_row\(\)](#), while it provides a significant added value.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*result\_type*

A constant that can take the following values: **MSQL\_ASSOC**, **MSQL\_NUM**, and **MSQL\_BOTH** with **MSQL\_BOTH** being the default.

## Return Values

Returns an array that corresponds to the fetched row, or **FALSE** if there are no more rows.

## Examples

### Example #2 - [mysql\\_fetch\\_array\(\)](#) example

```
<?php
$con = mysql_connect();
if (!$con) {
    die('Server connection problem: ' . mysql_error());
}

if (!mysql_select_db('test', $con)) {
    die('Database connection problem: ' . mysql_error());
}

$result = mysql_query('SELECT id, name FROM people', $con);
if (!$result) {
    die('Query execution problem: ' . mysql_error());
}
```

```
}

while ($row = msql_fetch_array($result, MSQL_ASSOC)) {
    echo $row['id'] . ': ' . $row['name'] . "\n";
}

msql_free_result($result);
?>
```

## ChangeLog

Version	Description
4.3.11 and 5.0.4	A bug was fixed when retrieving data from columns containing <b>NULL</b> values. Such columns were not placed into the resulting array.

## See Also

- [msql\\_fetch\\_row\(\)](#)
- [msql\\_fetch\\_object\(\)](#)
- [msql\\_data\\_seek\(\)](#)
- [msql\\_result\(\)](#)

# mysql\_fetch\_field

mysql\_fetch\_field -- Get field information

## Description

object **mysql\_fetch\_field** ( resource \$result [, int \$field\_offset ] )

[mysql\\_fetch\\_field\(\)](#) can be used in order to obtain information about fields in a certain query result.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*field\_offset*

The field offset. If not specified, the next field that wasn't yet retrieved by [mysql\\_fetch\\_field\(\)](#) is retrieved.

## Return Values

Returns an object containing field information. The properties of the object are:

- name - column name
- table - name of the table the column belongs to
- not\_null - 1 if the column cannot be **NULL**
- unique - 1 if the column is a unique key
- type - the type of the column

## See Also

- [mysql\\_field\\_seek\(\)](#)

# mysql\_fetch\_object

mysql\_fetch\_object -- Fetch row as object

## Description

object **mysql\_fetch\_object** ( resource \$result )

[mysql\\_fetch\\_object\(\)](#) is similar to [mysql\\_fetch\\_array\(\)](#), with one difference - an object is returned, instead of an array. Indirectly, that means that you can only access the data by the field names, and not by their offsets (numbers are illegal property names).

Speed-wise, the function is identical to [mysql\\_fetch\\_array\(\)](#), and almost as quick as [mysql\\_fetch\\_row\(\)](#) (the difference is insignificant).

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

## Return Values

Returns an object with properties that correspond to the fetched row, or **FALSE** if there are no more rows.

## Examples

### Example #3 - [mysql\\_fetch\\_object\(\)](#) example

```
<?php
$con = mysql_connect();
if (!$con) {
    die('Server connection problem: ' . mysql_error());
}

if (!mysql_select_db('test', $con)) {
    die('Database connection problem: ' . mysql_error());
}

$result = mysql_query('SELECT id, name FROM people', $con);
if (!$result) {
    die('Query execution problem: ' . mysql_error());
}

while ($row = mysql_fetch_object($result, MYSQL_ASSOC)) {
    echo $row->id . ': ' . $row->name . "\n";
}
```

```
}  
  
mysql_free_result($result);  
?>
```

## ChangeLog

Version	Description
4.3.11 and 5.0.4	A bug was fixed when retrieving data from columns containing <b>NULL</b> values. Such columns were not placed into the resulting array.

## See Also

- [mysql\\_fetch\\_array\(\)](#)
- [mysql\\_fetch\\_row\(\)](#)
- [mysql\\_data\\_seek\(\)](#)
- [mysql\\_result\(\)](#)

# mysql\_fetch\_row

mysql\_fetch\_row -- Get row as enumerated array

## Description

array **mysql\_fetch\_row** ( resource \$result )

[mysql\\_fetch\\_row\(\)](#) fetches one row of data from the result associated with the specified query identifier. The row is returned as an array. Each result column is stored in an array offset, starting at offset 0.

Subsequent call to [mysql\\_fetch\\_row\(\)](#) would return the next row in the result set, or **FALSE** if there are no more rows.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

## Return Values

Returns an array that corresponds to the fetched row, or **FALSE** if there are no more rows.

## Examples

### Example #4 - [mysql\\_fetch\\_row\(\)](#) example

```
<?php
$con = mysql_connect();
if (!$con) {
    die('Server connection problem: ' . mysql_error());
}

if (!mysql_select_db('test', $con)) {
    die('Database connection problem: ' . mysql_error());
}

$result = mysql_query('SELECT id, name FROM people', $con);
if (!$result) {
    die('Query execution problem: ' . mysql_error());
}

while ($row = mysql_fetch_row($result)) {
    echo $row[0] . ': ' . $row[1] . "\n";
}
```

```
mysql_free_result($result);  
?>
```

## ChangeLog

Version	Description
4.3.11 and 5.0.4	A bug was fixed when retrieving data from columns containing <b>NULL</b> values. Such columns were not placed into the resulting array.

## See Also

- [mysql\\_fetch\\_array\(\)](#)
- [mysql\\_fetch\\_object\(\)](#)
- [mysql\\_data\\_seek\(\)](#)
- [mysql\\_result\(\)](#)



# mysql\_field\_flags

mysql\_field\_flags -- Get field flags

## Description

string **mysql\_field\_flags** ( resource \$result, int \$field\_offset )

[mysql\\_field\\_flags\(\)](#) returns the field flags of the specified field.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*field\_offset*

The numerical field offset. The *field\_offset* starts at 1.

## Return Values

Returns a string containing the field flags of the specified key. This can be: *primary key not null*, *not null*, *primary key*, *unique not null* or *unique*.

# mysql\_field\_len

mysql\_field\_len -- Get field length

## Description

int **mysql\_field\_len** ( resource \$result, int \$field\_offset )

[mysql\\_field\\_len\(\)](#) returns the length of the specified field.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*field\_offset*

The numerical field offset. The *field\_offset* starts at 1.

## Return Values

Returns the length of the specified field or **FALSE** on error.

# mysql\_field\_name

mysql\_field\_name -- Get the name of the specified field in a result

## Description

string **mysql\_field\_name** ( resource \$result, int \$field\_offset )

[mysql\\_field\\_name\(\)](#) gets the name of the specified field index.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*field\_offset*

The numerical field offset. The *field\_offset* starts at 1.

## Return Values

The name of the field or **FALSE** on failure.

## See Also

- [mysql\\_field\\_len\(\)](#)

# mysql\_field\_seek

mysql\_field\_seek -- Set field offset

## Description

bool **mysql\_field\_seek** ( resource \$result, int \$field\_offset )

Seeks to the specified field offset. If the next call to [mysql\\_fetch\\_field\(\)](#) won't include a field offset, this field would be returned.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*field\_offset*

The numerical field offset. The *field\_offset* starts at 1.

## Return Values

Returns **TRUE** on success or **FALSE** on failure.

## See Also

- [mysql\\_fetch\\_field\(\)](#)

# mysql\_field\_table

mysql\_field\_table -- Get table name for field

## Description

int **mysql\_field\_table** ( resource \$result, int \$field\_offset )

Returns the name of the table that the specified field is in.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*field\_offset*

The numerical field offset. The *field\_offset* starts at 1.

## Return Values

The name of the table on success, or **FALSE** on failure.

# mysql\_field\_type

mysql\_field\_type -- Get field type

## Description

string **mysql\_field\_type** ( resource \$result, int \$field\_offset )

[mysql\\_field\\_type\(\)](#) gets the type of the specified field index.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*field\_offset*

The numerical field offset. The *field\_offset* starts at 1.

## Return Values

The type of the field. One of *int*, *char*, *real*, *ident*, *null* or *unknown*. This functions will return **FALSE** on failure.

# mysql\_fieldflags

mysql\_fieldflags -- Alias of [mysql\\_field\\_flags\(\)](#)

## Description

This function is an alias of [mysql\\_field\\_flags\(\)](#).

# mysql\_fieldlen

mysql\_fieldlen -- Alias of [mysql\\_field\\_len\(\)](#)

## Description

This function is an alias of [mysql\\_field\\_len\(\)](#).



# mysql\_fieldname

mysql\_fieldname -- Alias of [mysql\\_field\\_name\(\)](#)

## Description

This function is an alias of [mysql\\_field\\_name\(\)](#).

# mysql\_fieldtable

mysql\_fieldtable -- Alias of [mysql\\_field\\_table\(\)](#)

## Description

This function is an alias of [mysql\\_field\\_table\(\)](#).

# mysql\_fieldtype

mysql\_fieldtype -- Alias of [mysql\\_field\\_type\(\)](#)

## Description

This function is an alias of [mysql\\_field\\_type\(\)](#).

# mysql\_free\_result

mysql\_free\_result -- Free result memory

## Description

bool **mysql\_free\_result** ( resource \$result )

[mysql\\_free\\_result\(\)](#) frees the memory associated with *query\_identifier*. When PHP completes a request, this memory is freed automatically, so you only need to call this function when you want to make sure you don't use too much memory while the script is running.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

## Return Values

Returns **TRUE** on success or **FALSE** on failure.

# mysql\_list\_dbs

mysql\_list\_dbs -- List mSQL databases on server

## Description

resource **mysql\_list\_dbs** ( [ resource *\$link\_identifier* ] )

[mysql\\_list\\_tables\(\)](#) lists the databases available on the specified *link\_identifier*.

## Parameters

*link\_identifier*

The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns a result set which may be traversed with any function that fetches result sets, such as [mysql\\_fetch\\_array\(\)](#). On failure, this function will return **FALSE**.

## See Also

- [mysql\\_list\\_tables\(\)](#)
- [mysql\\_list\\_fields\(\)](#)

# mysql\_list\_fields

mysql\_list\_fields -- List result fields

## Description

resource **mysql\_list\_fields** ( string \$database, string \$tablename [, resource \$link\_identifier ] )

[mysql\\_list\\_fields\(\)](#) returns information about the given table.

## Parameters

*database*

The name of the database.

*tablename*

The name of the table.

*link\_identifier*

The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns a result set which may be traversed with any function that fetches result sets, such as [mysql\\_fetch\\_array\(\)](#). On failure, this function will return **FALSE**.

## See Also

- [mysql\\_list\\_tables\(\)](#)
- [mysql\\_list\\_dbs\(\)](#)

# mysql\_list\_tables

mysql\_list\_tables -- List tables in an mSQL database

## Description

resource **mysql\_list\_tables** ( string \$database [, resource \$link\_identifier ] )

[mysql\\_list\\_tables\(\)](#) lists the tables on the specified *database*.

## Parameters

*database*

The name of the database.

*link\_identifier*

The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns a result set which may be traversed with any function that fetches result sets, such as [mysql\\_fetch\\_array\(\)](#). On failure, this function will return **FALSE**.

## See Also

- [mysql\\_list\\_fields\(\)](#)
- [mysql\\_list\\_dbs\(\)](#)

# mysql\_num\_fields

mysql\_num\_fields -- Get number of fields in result

## Description

int **mysql\_num\_fields** ( resource *\$result* )

[mysql\\_num\\_fields\(\)](#) returns the number of fields in a result set.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

## Return Values

Returns the number of fields in the result set.

## See Also

- [mysql\\_query\(\)](#)
- [mysql\\_fetch\\_field\(\)](#)
- [mysql\\_num\\_rows\(\)](#)



# msql\_num\_rows

msql\_num\_rows -- Get number of rows in result

## Description

int **msql\_num\_rows** ( resource \$query\_identifier )

[msql\\_num\\_rows\(\)](#) returns the number of rows in a result set.

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [msql\\_query\(\)](#).

## Return Values

Returns the number of rows in the result set.

## See Also

- [msql\\_num\\_fields\(\)](#)

# mysql\_numfields

mysql\_numfields -- Alias of [mysql\\_num\\_fields\(\)](#)

## Description

This function is an alias of [mysql\\_num\\_fields\(\)](#).

# mysql\_numrows

mysql\_numrows -- Alias of [mysql\\_num\\_rows\(\)](#)

## Description

This function is an alias of [mysql\\_num\\_rows\(\)](#).

# mysql\_pconnect

mysql\_pconnect -- Open persistent mSQL connection

## Description

resource **mysql\_pconnect** ( [ string \$hostname ] )

[mysql\\_pconnect\(\)](#) acts very much like [mysql\\_connect\(\)](#) with two major differences.

First, when connecting, the function would first try to find a (persistent) link that's already open with the same host. If one is found, an identifier for it will be returned instead of opening a new connection.

Second, the connection to the SQL server will not be closed when the execution of the script ends. Instead, the link will remain open for future use ( [mysql\\_close\(\)](#) will not close links established by this function).

## Parameters

*hostname*

The hostname can also include a port number. e.g. *hostname,port*. If not specified, the connection is established by the means of a Unix domain socket, being then more efficient than a localhost TCP socket connection.

Note
While this function will accept a colon (:) as a host/port separator, a comma (,) is the preferred method.

## Return Values

Returns a positive mSQL link identifier on success, or **FALSE** on error.

## See Also

- [mysql\\_connect\(\)](#)
- [mysql\\_close\(\)](#)

# mysql\_query

mysql\_query -- Send mSQL query

## Description

resource **mysql\_query** ( string *\$query* [, resource *\$link\_identifier* ] )

[mysql\\_query\(\)](#) sends a query to the currently active database on the server that's associated with the specified link identifier.

## Parameters

*query*  
The SQL query.

*link\_identifier*  
The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns a positive mSQL query identifier on success, or **FALSE** on error.

## Examples

### Example #5 - [mysql\\_query\(\)](#) example

```
<?php
$link = mysql_connect("dbserver")
    or die("unable to connect to mysql server: " . mysql_error());
mysql_select_db("db", $link)
    or die("unable to select database 'db': " . mysql_error());

$result = mysql_query("SELECT * FROM table WHERE id=1", $link);
if (!$result) {
    die("query failed: " . mysql_error());
}

while ($row = mysql_fetch_array($result)) {
    echo $row["id"];
}
?>
```

## See Also

- [msgl\\_db\\_query\(\)](#)
- [msgl\\_select\\_db\(\)](#)
- [msgl\\_connect\(\)](#)

# mysql\_regcase

mysql\_regcase -- Alias of [sql\\_regcase\(\)](#)

## Description

This function is an alias of [sql\\_regcase\(\)](#).

# mysql\_result

mysql\_result -- Get result data

## Description

string **mysql\_result** ( resource \$result, int \$row [, mixed \$field ] )

[mysql\\_result\(\)](#) returns the contents of one cell from a mSQL result set.

When working on large result sets, you should consider using one of the functions that fetch an entire row (specified below). As these functions return the contents of multiple cells in one function call, they are often much quicker than [mysql\\_result\(\)](#).

Recommended high-performance alternatives: [mysql\\_fetch\\_row\(\)](#), [mysql\\_fetch\\_array\(\)](#), and [mysql\\_fetch\\_object\(\)](#).

## Parameters

*result*

The result [resource](#) that is being evaluated. This result comes from a call to [mysql\\_query\(\)](#).

*row*

The row offset.

*field*

Can be the field's offset, or the field's name, or the field's table dot field's name (tablename.fieldname.). If the column name has been aliased ('select foo as bar from ...'), use the alias instead of the column name.

Note
Specifying a numeric field offset is much quicker than specifying a fieldname or tablename.fieldname.

## Return Values

Returns the contents of the cell at the row and offset in the specified mSQL result set.



# mysql\_select\_db

mysql\_select\_db -- Select mSQL database

## Description

bool **mysql\_select\_db** ( string \$database\_name [, resource \$link\_identifier ] )

[mysql\\_select\\_db\(\)](#) sets the current active database on the server that's associated with the specified *link\_identifier*.

Subsequent calls to [mysql\\_query\(\)](#) will be made on the active database.

## Parameters

*database\_name*

The database name.

*link\_identifier*

The mSQL connection. If not specified, the last link opened by [mysql\\_connect\(\)](#) is assumed. If no such link is found, the function will try to establish a link as if [mysql\\_connect\(\)](#) was called, and use it.

## Return Values

Returns **TRUE** on success or **FALSE** on failure.

## See Also

- [mysql\\_connect\(\)](#)
- [mysql\\_pconnect\(\)](#)
- [mysql\\_query\(\)](#)

# mysql\_tablename

mysql\_tablename -- Alias of [mysql\\_result\(\)](#)

## Description

This function is an alias of [mysql\\_result\(\)](#).

# mysql

mysql -- Alias of [mysql\\_db\\_query\(\)](#)

## Description

This function is an alias of [mysql\\_db\\_query\(\)](#).