

How to Convert MySQL to MySQLi

In PHP 7, the MySQL extension is completely removed. Learn how to easily convert a MySQL extension into MySQLi without it.

_by

[Saquib Rizwan](#)

.

Apr. 18, 17 · [Web Dev Zone](#) · Tutorial

Like [\(6\)](#)

Save

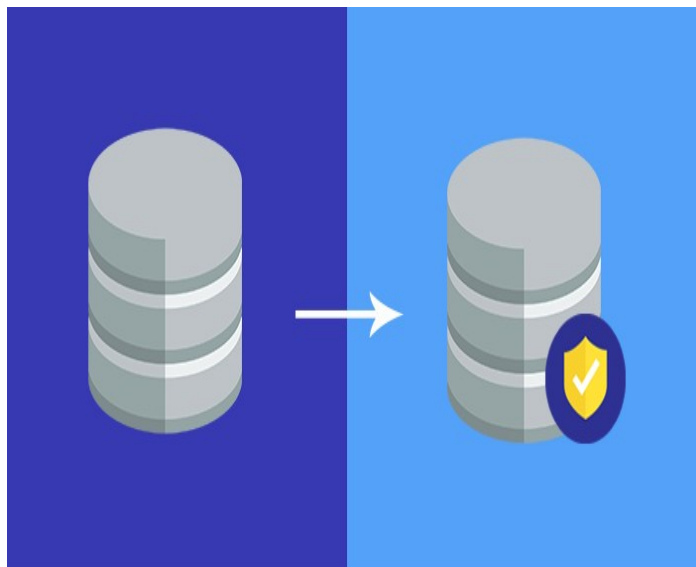
[Tweet](#)

115.71K Views

Join the DZone community and get the full member experience.

[Join For Free](#)

One of the most important developments in the PHP world was the backward compatibility break for the PHP MySQL extension, which leaves us with two methods to connect to the database: MySQLi and [PDO](#).



In PHP 7, the MySQL extension is completely removed. Thus, in this article, I will discuss how to convert a MySQL extension into MySQLi. The first thing you should understand is that MySQL works as a resource whereas MySQLi works as a resource and an object. While you really do not need to know the technical differences, you must understand that these two are a lot different from each other.

MySQL to MySQLi Procedural Methods

I will first start with the most commonly used MySQL methods. The only thing that changed in the following method is the addition of "i" in MySQL and some changes of the positions of `link` and `result` in the parameter of the method.

Following are the popularly used methods of MySQL, followed by the equivalent methods in MySQLi.

```
1
mysql_affected_rows -> mysqli_affected_rows($link)
2

3
mysql_close -> mysqli_close($link)
4

5
mysql_data_seek -> mysqli_data_seek( $result, $offset)
6

7
mysql_errno -> mysqli_errno( $link)
8

9
mysql_error -> mysqli_error( $link)
10

11
mysql_fetch_array -> mysqli_fetch_array( $result, $type)
```

If you want to convert your script from a MySQL extension to a MySQLi extension manually, you must start with the top of the script and start converting all the methods one by one.

For this, open the script in any text editor and use the find-and-replace tool to replace `mysql_` with `mysqli`. But that's not all. You must manually check and verify the parameters of the method, as well.

To automate the process of converting the script from MySQL to MySQLi, I can use the following online tools. I will create a sample insertion and retrieval code and convert it using one of these tools.

Sample Code for Conversion

I am going to use the following database and code for testing the online conversion tools.

Database SQL

```
1
CREATE TABLE `info`.`student`
2
3
(
```

```

        `id`          INT NOT NULL auto_increment,
4      `name`         VARCHAR(40) NOT NULL,
5      `last name`    VARCHAR(40) NOT NULL,
6      `email`        VARCHAR(40) NOT NULL,
7
      PRIMARY KEY (`id`)
8    )
9
engine = innodb;

```

Connection.php

```

1
<?php
2
$servername = "localhost" ;
3
$username = "root";
4
$pass = "";
5
$con = mysql_connect($servername,$username,$pass) or die("Problem occur in
connection");
6
$db = mysql_select_db("info",$con);
7
?>

```

Insert.php

```

1
<?php
2
require_once("connection.php");
3
$query = "Insert into student
values('','Saquib','Rizwan','saquib.rizwan@cloudways.com')";
4
$result = mysql_query($query);
5
if(isset($result))
6
{
7
    echo "Data has been inserted";
8
}
9
?>

```

Retrieve.php

```
1
<?php
2
require_once("connection.php");
3
$query = "select * from Student ";
4
$result = mysql_query($query);
5
while($data = mysql_fetch_array($result))
6
{
7
    echo "<br/><br/>";
8
    echo "ID : $data[0]";
9
    echo "Name : $data[1] ";
10
    echo "Last Name : $data[2] ";
11
    echo "Email : $data[3] ";
```

Here is the basic PHP code which uses MySQL extension to create the database connection, select database, create and execute a query and fetch data from the database.

Now I will convert the script using an online tool.

ICT Academie

In order to convert the code using [ICT Academie](#), all you need to do is paste your code snippet enclosed in PHP opening and closing tags.

For example, if you want to see how `mysql_connect()` gets converted, paste `<?php mysql_connect() ?>>`.

Code conversion result from the ICT Academie for the above code is as follows.

Connection.php

```
1
<?php
2
$servername = "localhost" ;
3
$username = "root";
4
$pass = "";
5
$con = ($GLOBALS["__mysqli_ston"] = mysqli_connect($servername, $username,
$pass)) or die("Problem occur in connection");
6
$db = ((bool)mysqli_query($con, "USE " . info));
```

7
?>

Insert.php

```
1
<?php
2
require_once("connection.php");
3
$query = "Insert into student
values('','Saquib','Rizwan','saquib.rizwan@cloudways.com')";
4
$result = mysqli_query($GLOBALS["__mysqli_ston"], $query);
5
if(isset($result))
6
{
7
    echo "Data has been inserted";
8
}
9
?>
```

Retrieve.php

```
1
<?php
2
require_once("connection.php");
3
$query = "select * from Student ";
4
$result = mysqli_query($GLOBALS["__mysqli_ston"], $query);
5
while($data = mysqli_fetch_array($result))
6
{
7
    echo "<br/><br/>";
8
    echo "ID : $data[0]    ";
9
    echo "Name : $data[1]  ";
10
    echo "Last Name : $data[2]  ";
11
    echo "Email : $data[3]  ";
```

To test the converted code, it is clear that all the MySQL methods have been successfully converted into MySQLi. It only changes the PHP code, but the result remains the same.

Note: Before starting the conversion process, create backups of all the files and test run all the code before live deployment. The converter only converts the code. It does not add any security measures to prevent SQL injections.

<https://dzone.com/articles/convert-mysql-to-mysqli>