

Nach einem erfolgreichen Verbindungsaufbau zur Datenbank wollen wir selbstverständlich auch SQL Befehle abfragen. Dafür gibt es die `mysqli_query()` Funktion, die den Datenbank-Handler und den SQL-Code als String benötigt. Mit `mysqli_fetch_all()` wird dann das Ergebnis der Abfrage in ein Array gespeichert.

PHP

`mysqli_query()`

Zuerst wird mit `mysqli_connect` die Verbindung zur Datenbank aufgebaut. Das Datenbankobjekt wird der Variable `$db` übergeben.

Im zweiten Schritt wird die SQL Abfrage (die Spalten FunkID und Firmenname der Tabelle Medienbetriebe) als String in der Variable `$sql` gespeichert.

Die `mysqli_query($db, $sql)` Methode benötigt den Datenbankhandler (`$db`) und den SQL Code (`$sql`). Das Ergebnis der Abfrage wird in der Variable `$ergebnis` gespeichert.

Zuletzt wird das Ergebnis mit `mysqli_fetch_all()` ausgewertet und im `$ausgabeArray` gespeichert.

Am Schluss sollte man die Datenbankverbindung mit `mysqli_close()` wieder schließen.

```
<?php
    $db = mysqli_connect("localhost", "MedienAdmin",
                        "geheim", "MedienDB");

    $sql = "SELECT FunkID, Firmenname FROM Medienbetriebe";
    $ergebnis = mysqli_query($db, $sql);

    $ausgabeArray = mysqli_fetch_all($ergebnis, MYSQLI_ASSOC);
    mysqli_close($db);
?>
```



Der zweite Parameter der `mysqli_fetch_all` Funktion definiert die Art der Rückgabe, also wie das Array aufgebaut sein soll:

`MYSQLI_ASSOC` Assoziiertes Array (mit Namen für die Schlüssel)
`MYSQLI_NUM` Numerisches Array (mit aufsteigenden Zahlen als Schlüssel)
`MYSQLI_BOTH` Beide Arten für die Schlüssel

Mit `print_r($ausgabeArray)` kann man sich das Array ansehen!



Das Array kann dann mit einer Schleife weiter verarbeitet werden. Im Beispiel unten werden die Datensätze in einer Aufzählung ausgegeben!

```
echo '<ul>';
for($i = 0; $i < count($ausgabeArray); $i++) {
    echo '<li>' . $ausgabeArray[$i]["FunkID"] . ' - ';
    echo $ausgabeArray[$i]["Firmenname"] . '</li>';
}
echo '</ul>';
```