

Tips voor het werken met PHP en MySQL

Voor het bouwen van een website met een MySQL database heb je PHP nodig. Het maken van de database kan gedaan worden met PHP scripts of met een .sql file script. In deze les wordt naar beide mogelijkheden gekeken. Tevens wordt er getoond hoe je MAMP zo kunt configureren dat je PHP error meldingen kunt zien in de browser.

PHP scripts

Door gebruik te maken van PHP scripts. Deze zijn eenvoudig te kopiëren van <https://www.w3schools.com/php/default.asp>. Ga op de pagina in het menu gedeelte (links) een beetje naar onderen tot je bij **MySQL Database** komt. Zie afbeelding hier onder:

Hier kun je alle scripts vinden die je nodig zult hebben voor het werken met de database. Er zijn een paar scripts die je zeker gemaakt moet hebben voor het project.

connectDB.php

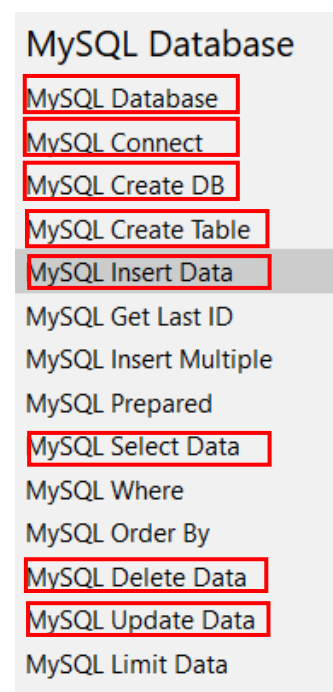
Hier zit de connectie script in om verbinding te maken met jullie database in MySQL.

createDB.php

Hier wordt de database mee aangemaakt in MySQL door het script

createTBLS.php

Hier worden de tabellen in jullie database aan gemaakt in MySQL.



Verder zullen jullie PHP scripts hebben waarin je informatie gaat opvragen uit de database, toevoegt aan de database, aanpast of verwijderd uit de database.

De eerste drie scripts kun je bijna letterlijk van w3schools.com overnemen. Alleen jullie eigen relevante gegevens dienen jullie te wijzigen.

Voor de connectie script zijn er twee opties MySQLi en PDO. Beiden zijn bruikbaar. In Fundament wordt voornamelijk PDO gebruikt, dit is een meer algemene manier om via PHP verbinding te maken met een MySQL database. MySQLi is speciaal gemaakt om met MySQL te werken terwijl PDO met verschillende databases kan werken. Opzicht maakt het niet uit welk van de twee je gebruikt, maar je moet ze niet doorelkaar gebruiken. Daar we met een MySQL database werken zou ik zeggen gebruik MySQLi.

Hieronder zie je het connectie script zoals die in w3schools.com staat om verbinding te maken met de server en je database:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Create connection
$conn = new mysqli($servername, $username, $password);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Create database
$sql = "CREATE DATABASE myDB";
if ($conn->query($sql) === TRUE) {
    echo "Database created successfully";
} else {
    echo "Error creating database: " . $conn->error;
}

$conn->close();
?>
```

Het enige dat jullie in dit script hoeven te veranderen is wat er is ingevuld bij de variabelen \$username en \$password. In MAMP zijn deze beide standaard 'root'. En in de variabele \$sql moet je de myDB vervangen voor de naam van de database die jullie hebben gekozen.

OPMERKING: Als je de code van hier kopieert moet je wel de dubbele aanhalingstekens even op nieuw typen daar Word er smart quotes (") van maakt en die kunnen niet door PHP gebruikt worden. Het is dus verstandiger om het script van w3schools.com te kopiëren naar notepad++.

Wanneer je dit script uitvoert vanuit je browser zullen jullie in de MySQL interface van MAMP aan de linker kant in de kolom met database namen zien dat jullie database is toegevoegd.

Na het uitvoeren van dit script kunnen jullie een script waarin de tabellen staan uitvoeren. Deze is ook eenvoudig te kopiëren van w3schools.com. Voor het maken van de tabellen zul je eerst de tabellen moeten hebben gedefinieerd in SQL. Dit wil zeggen de naam van de tabel, de velden met hun datatype en lengte moet bekend zijn. Ook moet bekend zijn welk(e)

veld(en) zullen dienen al primaire sleutels of secundaire (vreemde) sleutels voor de eventuele relaties tussen de tabellen. In w3schools.com wordt dit allemaal opnieuw uitgelegd met voorbeelden.

Hieronder zien jullie een voorbeeld van het script om een tabel te maken in jullie database zoals deze staat op de website w3schools.com.

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// sql to create table
$sql = "CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP
)";

if ($conn->query($sql) === TRUE) {
    echo "Table MyGuests created successfully";
} else {
    echo "Error creating table: " . $conn->error;
}

$conn->close();
?>
```

Dit script maakt 1 tabel aan maar is eenvoudig aan te passen om meerdere tabellen aan te kunnen maken. De variabelen \$servername, \$username, \$password en \$dbname zullen door jullie moeten worden voorzien van de informatie voor jullie project. In de variabele \$sql moet de informatie komen te staan om jullie tabel aan te maken in MySQL zoals dat nodig is voor jullie project. Bij de echo in de if-functie onder het script moet je de naam ook aanpassen.

Jullie kunnen deze twee stukken, zonder de **\$conn->close();** meerdere keren herhalen om de andere tabellen aan te maken in dit zelfde script. Wanneer jullie dit script uitvoeren zullen jullie zien dat onder de database naam in het MySQL interface nu ook de tabellen verschijnen.

De database is nu klaar voor gebruik. De instructies die hier gebruikt werden behoren tot het DDL del van MySQL.

Eerder verteld dat er een tweede manier is om dit te doen en dat is door een .sql bestand te maken met notepad++. Voordeel van deze manier is dat je niet te maken hebt met PHP code en minder files nodig hebt. Dus eenvoudiger.

Tevens kunnen jullie in deze file ook meteen test data toevoegen om te kijken of jullie site werkt zoals dat gepland was. Dit gebeurt met de instructie:

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

Of als je waarden aan alle kolommen gaat toevoegen kun je gebruik maken van de kortere versie:

```
INSERT INTO table_name
VALUES (value1, value2, value3, ...);
```

Hierbij moet natuurlijk voor *table_name, column1, etc.* jullie data worden ingevuld. Hier naast een voorbeeld van hoe zo'n .sql bestand eruit zou kunnen zien:

```
1 CREATE DATABASE testdb;
2
3 USE testdb;
4
5 CREATE TABLE MyGuests (
6   id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
7   firstname VARCHAR(30) NOT NULL,
8   lastname VARCHAR(30) NOT NULL,
9   email VARCHAR(50));
10
11 INSERT INTO MyGuests
12 VALUES (1111, 'Pedro', 'Picapiedra', 'PP@gmail.com');
```

Voor het testen van de site zullen jullie vaker de database moeten verwijderen en opnieuw aan maken. Dit kan dus ook eenvoudig met een .sql file door één regel aan het begin toe te voegen zoals in de afbeelding hier naast (kijk naar regel 1):

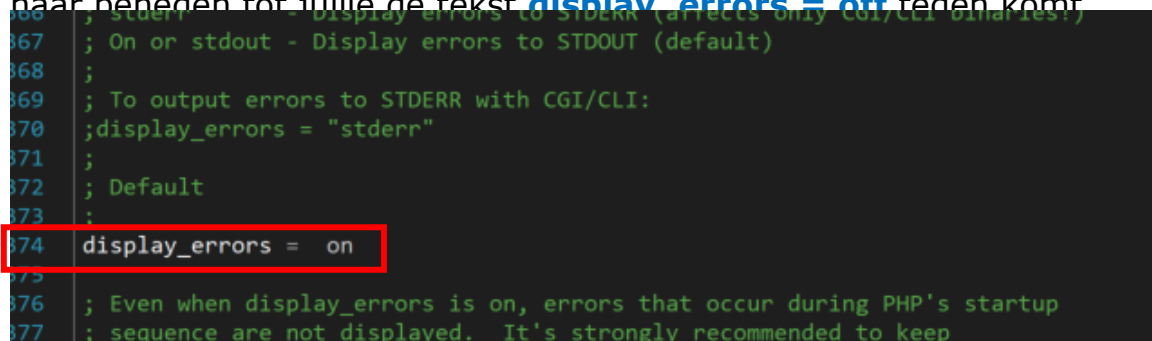
```
1 DROP DATABASE IF EXISTS testdb;
2
3 CREATE DATABASE testdb;
4
5 USE testdb;
6
7 CREATE TABLE MyGuests (
8   id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
9   firstname VARCHAR(30) NOT NULL,
10  lastname VARCHAR(30) NOT NULL,
11  email VARCHAR(50));
12
13 INSERT INTO MyGuests
14 VALUES (1111, 'Pedro', 'Picapiedra', 'PP@gmail.com');
```

Het zal duidelijk zijn dat het werken met één .sql file die je iedere keer opnieuw kunt importeren en uitvoeren in MAMP het werken aan het project eenvoudiger en sneller zal laten verlopen.

PHP errors

Jullie zullen merken of misschien al gemerkt hebben dat als je PHP scripts uitvoert in MAMP er geen foutmeldingen getoond worden. Er wordt alleen een blanco wit scherm getoond. Dit kun je aan passen door in de php.ini file een setting aan te passen.

Wanneer je MAMP op de standaard manier hebt geïnstalleerd kunnen jullie het **php.ini** bestand vinden in de map **/MAMP/conf/php7.0.9/php.ini**. Open dit bestand in notepad++ en ga naar beneden tot jullie de tekst **display_errors = off** tegen komt



```
366 ; stderr - Display errors to STDERR (affects only CGI/CLI binaries)
367 ; On or stdout - Display errors to STDOUT (default)
368 ;
369 ; To output errors to STDERR with CGI/CLI:
370 ;display_errors = "stderr"
371 ;
372 ; Default
373 ;
374 display_errors = on
375 ;
376 ; Even when display_errors is on, errors that occur during PHP's startup
377 ; sequence are not displayed. It's strongly recommended to keep
```

Verander de off in on en sla het bestand op. Als je MAMP gestart had sluit het dan af en start het opnieuw op. Nu zal MAMP de PHP errors wel tonen in het venster van je browser.