


07 – Grillparty Teil 2

Im zweiten Teil wird die Webseite um das Eingabeformular ergänzt, so dass sich weitere Freunde von Anna und Thilo für die Grillparty anmelden können.

Legen Sie also zunächst unterhalb der Tabelle ein Formular mit den drei Eingabefeldern für Vorname, Nachname und Beitrag an.



Einladung zur Grillparty

Wir planen eine große Grillparty - sei dabei!

Hallo, wir planen eine große Grillparty. Es gibt reichlich Essen, Trinken und Musik. Die Feier soll am Samstag in zwei Wochen ab 15 Uhr bei uns im Garten stattfinden. Ihr seid herzlich eingeladen. Wir würden uns sehr über Deine/Eure Zusage freuen. Außerdem fänden wir es toll, wenn jeder eine Kleinigkeit zum Büffet beitragen könnte. (Grillgut, Salat, Brot, Dips, Nachtsch, ...). Grill und Getränke besorgen wir! Bitte tragt euch in die Liste ein!

Liebe Grüße, Anna und Thilo.

Bisherige Zusagen:

Nummer	Vorname	Nachname	Ich bringe mit
1	Florian	Bertel	Nudelsalat
2	Kerstin	Janke	Herrencreme
3	Robert	Wagner	10 Bratwürstchen

Trage Dich in die Liste ein!

Vorname:

Nachname:

Ich bringe mit:

Daten einfügen

Beginnen Sie die PHP-Anweisungen mit der bekannten if-Abfrage, die prüft, ob der Submit-Button gedrückt wurde. Sie können den PHP-Bereich zum Beispiel direkt unterhalb der Tabelle erstellen.

Letzten Endes ist es aber egal, an welcher Stelle im Dokument der PHP-Code steht.

```
<?php
    <input id = "submit" type = "submit" name = "submit">
  </form>

  if(isset($_POST["submit"]))
  {
```

Die grundlegende Funktion, um SQL-Anweisungen zum Datenbankserver zu schicken, haben Sie bereits kennengelernt.

mysqli_query(\$con, \$sql);

Im ersten Teil ging es darum, Daten **aus** der Datenbank mit dem SELECT-Befehl auszulesen, jetzt sollen Daten **in** die Tabelle eingefügt werden. Der Befehl dazu lautet:

INSERT INTO gäste (vorname, nachname, beitrag) VALUES ('...', '...', '...')

Bzw.:

INSERT INTO gäste VALUES (default, '...', '...')

Als Values werden die Werte aus dem Formular übergeben. Nach dem Absenden befinden sich die Eingaben in der Variablen `$_POST`.

Achten Sie bei der SQL-Anweisung auf die doppelten und einfachen Anführungszeichen.

```
$sql = "INSERT INTO gäste VALUES('default','".$_POST["vorname"]."', '".$_POST["nachname"]."', '".$_POST["beitrag"]."')";
```

- Die einfachen Hochkommata werden für den eigentlichen SQL-Befehl benötigt.
- Die doppelten Anführungszeichen unterbrechen bzw. beginnen die PHP-Zeichenkette, die der Variablen `$sql` übergeben wird.

Die Werte der `$_POST`-Variablen können auch zuvor in neue Variablen abgelegt werden.

```
$vname = $_POST["vorname"];  
$nname = $_POST["nachname"];  
$beitrag = $_POST["beitrag"];  
  
$sql = "INSERT INTO gäste VALUES('default','".$vname."', '".$_POST["nachname"]."', '".$_POST["beitrag"]."')";
```

Nachdem die Funktion **mysqli_query()** ausgeführt wurde (hier ist kein Rückgabewert vorhanden), kann mit Hilfe der Funktion **mysqli_affected_rows()** geprüft werden, ob der neue Datensatz erfolgreich eingefügt wurde.

Als Parameter wird der Funktion die Datenbank-Verknüpfung **\$con** übergeben. Es geht um die Anzahl der Zeilen, die vom vorherigen SQL-Befehl betroffen sind.

```
mysqli_query($con, $sql);

$num = mysqli_affected_rows($con);

if($num > 0)
    echo "<p><b>Du hast dich erfolgreich in die Liste eingetragen. Wir freuen uns auf Dich/Euch.</b></p>";
else
    echo "<p><b>Es gab anscheinend einen Fehler! Prüfe bitte, ob du in die Liste aufgenommen wurdest.</b></p>";
}
```

Bei Erfolg gibt die Funktion den Wert 1 zurück und bestätigt somit, dass 1 neuer Datensatz hinzugefügt wurde. An dieser Stelle sollte mit geeigneten Fehler- bzw. Erfolgsmeldungen gearbeitet werden (siehe if-else-Anweisung).

Das wars!

Testen Sie nun Ihre Webanwendung.

Wenn alles richtig funktioniert, sollte dies mit der entsprechenden Meldung bestätigt werden.

Senden

Du hast dich erfolgreich in die Liste eingetragen. Wir freuen uns auf Dich/Euch.

Laden Sie die Webseite erneut, um den neuen Eintrag auch in der Tabelle sehen zu können.