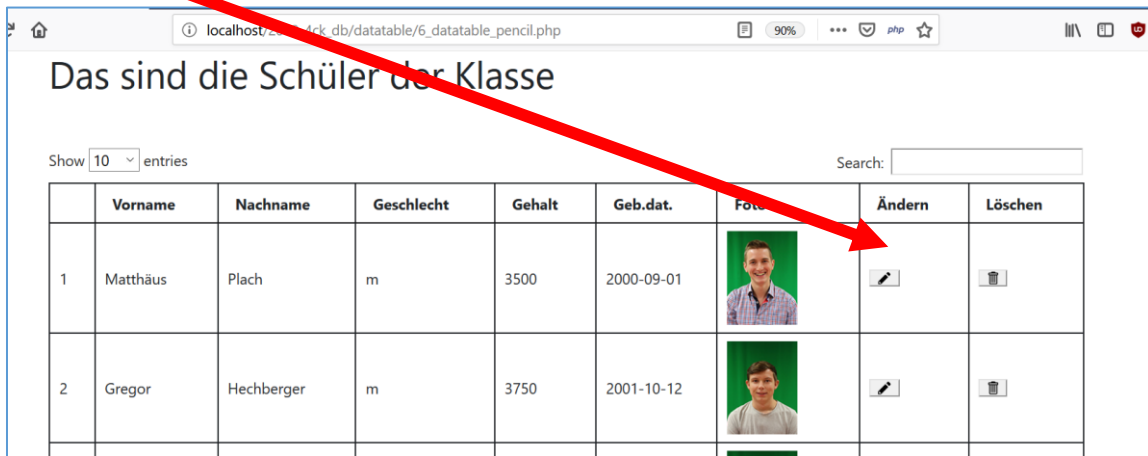








Datatable mit Buttons zum Bearbeiten und Löschen

Ziel:



Das sind die Schüler der Klasse

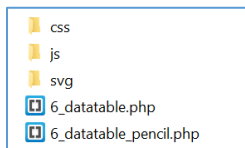
Show 10 entries Search:

	Vorname	Nachname	Geschlecht	Gehalt	Geb.dat.	Foto	Ändern	Löschen
1	Matthäus	Plach	m	3500	2000-09-01			
2	Gregor	Hechberger	m	3750	2001-10-12			

1) Bilder und Button einfügen

Öffne die Datei „datatable.php“ und speichere sie als „datatable_pencil.php“, mit der wir weiterarbeiten.

Kopiere zuerst den Ordner „svg“ zu den anderen bereits bestehenden Ordnern, damit die Bilder für den Bleistift und Papierkorb vorhanden sind.



1) Da durch den Klick auf die Buttons dann eine Action von einer Form benötigt wird, öffne und schließe nach dem Ende der <table> mit echo die <form>:

```

30 $erg = $con->query("SELECT * FROM schueler");
31
32 echo '<form method="post">';
33
34 echo "<table border='1' id='myTable' class='tablesorter"
35

```

```

48 }
49 echo "</table>";
50 echo "</form>";
51 ?>

```

2) In den Header sollen auch beide Elemente aufgenommen werden, nämlich „ändern“ und „löschen“.

```

34 echo "<table border='1' id='myTable' class='tablesorter'>";
35
36 echo "<thead><tr><th></th><th>Vorname</th><th>Nachname</th><th>Geschlec
</th><th>Foto</th><th>Ändern</th><th>Löschen</th></tr></thead>";
37 while($zeile = $erg->fetch_array()){

```

3) Erstelle in der „while“ aber noch vor dem ersten „echo“ eine Variable „\$zahl“ aus der ID. Das soll die ID dieser Zeile holen, damit er diese ID bei einem Klick auf einen der Buttons auch mitgegeben werden kann. Es soll ja genau diese Zeile später geändert oder gelöscht werden.

```

37 while($zeile = $erg->fetch_array()){
38     $zahl = $zeile['s_id'];
39     echo "<tr>|
40         <td>{$zeile['s_id']}</td>

```

4) Die beiden Buttons sollen nach dem Bild angeboten werden. Diese müssen in einem eigenen echo erstellt werden. Daher muss das „echo“ von vorne (Zeile 46) noch geschlossen werden, mit Strichpunkt nach dem </td> und den Anführungszeichen, die auch noch geschlossen werden müssen.

```

46     $zeile['foto'] )." ' /> </td>";
47     echo "<td><button type='submit' formaction='aendern.php' name='auswahl' value='". $zahl. "'>
48         <img src='svg/pencil.svg'></button></td>";
49     echo "<td><button type='submit' formaction='loeschen.php' name='auswahl' value='". $zahl. "'>
50         <img src='svg/trashcan.svg'></button></td>";
51     echo "</tr>";
52 }
53 echo "</table>";

```

Code:

```

echo "<td><button type='submit' formaction='aendern.php' name='auswahl' value='". $zahl. "'>
    <img src='svg/pencil.svg'></button></td>";

```

Beachte im value die unterschiedlichen Anführungszeichen und die Verknüpfungspunkte:

```

value='". $zahl. "'>

```

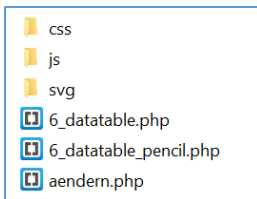
Info:

Ein Button ermöglicht, dass durch Draufklicken die „formaction“ ausgeführt werden kann und somit die andere PHP-Datei geöffnet wird

Der Button beinhaltet einen Type Submit, der auch einen „name“ hat und einen „value“. Der „value“ wird benötigt, dass der name='auswahl' und somit dessen „value“ (die \$zahl) mit der ID der gewünschten Zeile übergeben werden kann. Diese soll ja nachdem der Button geklickt wurde, in der nächsten PHP-Datei (hier z.B aendern.php bzw. loeschen.php) weiterverwendet werden können (mit \$_POST[]).

2) Datei „aendern.php“ erstellen

Kopiere die gerade erstellte Datei `6_datatable_pencil.php` und speichere sie als „aendern.php“.



Zuerst kann man die `<h1>` korrigieren:

```
10 <div class="container">
11 <h1>Bitte ändern Sie die falschen Daten und drücken dann auf Änderungen absenden</h1>
12 <br><br>
```

Diese muss nun um einige Zeilen gelöscht werden, da man ja keine „<table>“ benötigt, aber ein Formular. Dieses soll genau jene Daten von dem ausgewählten Datensatz enthalten, die man auch ändern können soll.

Lösche daher folgenden Teil, wo die Table beginnt und auch wieder endet. Daher soll nur davor und danach die Zeilen mit der `<form>` überbleiben:

```
30 $erg = $con->query("SELECT * FROM schueler");
31
32 echo '<form method="post">';
33
34
35
36
37 |
38 echo "</form>";
39 ?>
40
41 </div>
42 </body>
43 </html>
```

Lösche auch die Elemente, die die „datatable“ betreffen, wie

Zeile 8 und die Zeilen 14-18:

```
7 <script src="js/bootstrap.min.js"></script>
8 <script src="js/datatables.js"></script>
9 </head>
10 <body>
11 <div class="container">
12 <h1>Das sind die Schüler der Klasse</h1>
13 <br><br>
14 <script>
15 $(document).ready(function(){
16 $('#myTable').dataTable();
17 });
18 </script>
19
20 <?php
```

- Die Form soll nun eine weitere action erhalten, auf die nach dem Klick „Absenden“ dieses Formulars weitergeleitet werden soll. Dieses nenne „aendern2.php“ und soll die neuen Änderungen noch mal anzeigen und in die Datenbank mittels „input“ schreiben.

```

26
27 echo '<form method="post" action="aendern2.php">';
28
29 while($row = mysqli_fetch_object($erg))
    
```

- Nun muss die ausgewählte ID aus der Tabelle, die mit Klick auf das Bleistiftsymbol ausgewählt wurde, übernommen werden. Dazu erstelle noch vor Beginn der „SELECT“-Abfrage eine neue Variable (z.B. id_aendern) und hole mit \$_POST den passenden name „auswahl“ ab. Der ist der „name“ im Button, der das Bleistift-symbol enthält.

```

14 <?php
15
16 $id_aendern = $_POST['auswahl'];
17
18 $con = new MySQLi("localhost", "root"
    
```

- Nun muss die SELECT angepasst werden, da in den Formularfeldern gleich die ausgewählten Daten stehen sollen, die zur ID passen. Diese Daten werden aus der Datenbank geholt. Dabei wird mit der Abfrage WHERE genau die ID übergeben, die in der Datei vorher ausgewählt wurde und hier der Variable „\$id_aendern“ zugewiesen wurde.

```

26
27 $erg = $con->query("SELECT * FROM schueler WHERE s_id='".$id_aendern.'" ");
28
29 echo '<form method="post" action="aendern2.php">';
    
```

- Darunter muss man nun in der „while-Schleife“ genau jenen Datensatz aus der Datenbank holen, der zur ID passt. Dies geschieht in einem schon ausgefüllten Formular.

Dabei muss man auf die Felder der Datenbank beachten:

Die SELECT-Elemente müssen genau dazu passen.

Daher:

Erstelle eine „echo“-Ausgabe mit typischen Formularfeldern wie <label> und <input>. Jedoch ist der „value“ hier mit den Daten der Datenbank zu füllen.

#	Name	Typ	Kollation
<input type="checkbox"/>	1 s_id	int(11)	
<input type="checkbox"/>	2 vorname	varchar(30)	utf8_general_ci
<input type="checkbox"/>	3 nachname	varchar(50)	utf8_general_ci
<input type="checkbox"/>	4 geschlecht	char(1)	utf8_general_ci
<input type="checkbox"/>	5 klasse	varchar(3)	utf8_general_ci
<input type="checkbox"/>	6 gehalt	double	
<input type="checkbox"/>	7 geburtsdatum	date	
<input type="checkbox"/>	8 foto	blob	

```
value="' . $row -> s_id . '"
```

Im „value“ wird somit die Query \$row genommen und mit dem „Pfeil“ auf den gewünschten „Name“ aus der Datenbank gezeigt, hier die „s_id“.

```

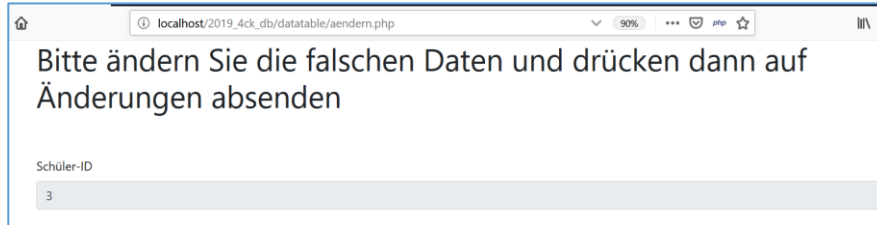
31 while($row = mysqli_fetch_object($erg))
32 {
33     echo '<label>Schüler-ID</label>
34         <input class="form-control" name="schuelerid" readonly value="' . $row -> s_id . ' ">';
35 };
36 echo '</form>';
37 ?>
    
```

Code:

```
echo '<label>Schüler-ID</label>'
    <input class="form-control" name="schuelerid" readonly value="".$row -> s_id.">;
```

Beachte: Hier wird das „readonly“ verwendet, da die ID nicht vom User geändert werden soll, sondern nur zur Info angezeigt wird.

Ergebnis:



- Nun folgen alle andern Felder, jedoch nicht mehr mit „readonly“.

Verwende alle korrekten „name“ aus der Datenbank.

#	name	Typ	Kollation
1	s_id	int(11)	
2	vorname	varchar(30)	utf8_general_ci
3	nachname	varchar(50)	utf8_general_ci
4	geschlecht	char(1)	utf8_general_ci
5	klasse	varchar(3)	utf8_general_ci
6	gehalt	double	
7	geburtsdatum	date	
8	foto	blob	

Code:

```
echo '<label>Vorname</label>'
    <input class="form-control" name="vorname"
value="".$row -> vorname."><br>;
    echo '<label>Nachname</label>'
    <input class="form-control" name="nachname"
value="".$row -> nachname."><br>;
    echo '<label>Geschlecht</label>'
    <input class="form-control" name="geschlecht" value="".$row -> geschlecht."><br>;
    echo '<label>Klasse</label>'
    <input class="form-control" name="klasse" value="".$row -> klasse."><br>;
    echo '<label>Gehalt</label>'
    <input class="form-control" name="gehalt" value="".$row -> gehalt."><br>;
    echo '<label>Geburtsdatum</label>'
    <input class="form-control" name="geburtsdatum" value="".$row -> geburtsdatum."><br>;
```

```
31 while($row = mysqli_fetch_object($erg))
32 {
33     echo '<label>Schüler-ID</label>'
34     <input class="form-control" name="schuelerid" readonly value="".$row -> s_id."><br>;
35     echo '<label>Vorname</label>'
36     <input class="form-control" name="vorname" value="".$row -> vorname."><br>;
37     echo '<label>Nachname</label>'
38     <input class="form-control" name="nachname" value="".$row -> nachname."><br>;
39     echo '<label>Geschlecht</label>'
40     <input class="form-control" name="geschlecht" value="".$row -> geschlecht."><br>;
41     echo '<label>Klasse</label>'
42     <input class="form-control" name="klasse" value="".$row -> klasse."><br>;
43     echo '<label>Gehalt</label>'
44     <input class="form-control" name="gehalt" value="".$row -> gehalt."><br>;
45     echo '<label>Geburtsdatum</label>'
46     <input class="form-control" name="geburtsdatum" value="".$row -> geburtsdatum."><br>;
47 };
```

Das Foto wird hier weggelassen.

Ergebnis:



localhost/2019_4ck_db/datatable/aendern.php 60%

Bitte ändern Sie die falschen Daten und drücken dann auf Änderungen absenden

Schüler-ID
1

Vorname
Matthäus

Nachname
Plach

Geschlecht
m

Klasse
4CK

Gehalt
3500

Geburtsdatum
2000-09-01

Bestätigen

Abbrechen

Die beiden Buttons darunter:

```

48
49     echo'<p><a href="aendern2.php"><input name="aendern2" class="btn btn-success btn-lg btn-block"
50     type="submit" value="Bestätigen"></a></p>';
51     echo'<p><a href ="6_datatable_pencil.php"><input name="back" class ="btn btn-danger btn-lg btn-block"
52     type="submit" value="Abbrechen"></a></p>';
53     echo '</form>';
54     ?>

```

Code:

```
echo'<p><a href="aendern2.php"><input name="aendern2" class="btn btn-success btn-lg btn-block"
type="submit" value="Bestätigen"></a></p>';
```

```
echo'<p><a href ="6_datatable_pencil.php"><input name="back" class ="btn btn-danger btn-lg btn-
block" type="submit" value="Abbrechen"></a></p>';
```

Info:

Der obere „submit-Button“ soll auf die nächste Seite „aendern2.php“ weiterleiten, der untere zurück zur Übersicht.

3) Site aendern2.php erstellen

- Klickt man in „aendern.php“ auf den Button „Bestätigen“ wird man auf die Site „aendern2.php“ weitergeleitet.
- Hier soll man eine Bestätigung der Bearbeitung und einen „zurück“-Button geliefert bekommen.
- Gleichzeitig soll die Änderung in die Dateien gespeichert werden.
- Die Übergabe der passenden ID erfolgt während die anderen „name“ auch übergeben werden, gleich zu Beginn des PHP-Scripts `16 $id = $_POST['schuelerid'];`

Speichere die soeben erstellte „aendern.php“ als „aendern2.php“. Hier soll nun weitergearbeitet werden.

Lösche das komplette <form>, sodass nur mehr folgendes im PHP-Teil übrig bleibt:

```

14 <?php
15
16 $id_aendern = $_POST['auswahl'];
17
18 $con = new MySQLi("localhost", "root", "", "schueler");
19 ▼ if ($con->connect_error) {
20     echo "Fehler bei der Verbindung: " . mysqli_connect_error();
21     exit();
22 }
23 ▼ if (!$con->set_charset("utf8")){
24     echo "Fehler bei utf8" . $con->error;
25 }
26
27 ?>

```

Die Verbindung in Zeile 18 und die IF in Zeile 23 sollen bleiben.

In Zeile 16 soll nun auch die \$id gelöscht werden, weil sie nicht benötigt wird.

- **Variablen erstellen**

Erstelle gleich nach dem Start der PHP-Struktur (<?php) die verschiedenen Variablen, die sich die Daten aus dem Formular aus „aendern.php“ holen soll.

```

14 <?php
15
16 $id = $_POST['schuelerid'];
17 $vor = $_POST['vorname'];
18 $nach = $_POST['nachname'];
19 $geschlecht = $_POST['geschlecht'];
20 $klasse = $_POST['klasse'];
21 $gehalt = $_POST['gehalt'];
22 $geb = $_POST['geburtsdatum'];

```

Code:

```

$id = $_POST['schuelerid'];
$vor = $_POST['vorname'];
$nach = $_POST['nachname'];
$geschlecht = $_POST['geschlecht'];
$klasse = $_POST['klasse'];
$gehalt = $_POST['gehalt'];
$geb = $_POST['geburtsdatum'];

```

Theorie: Aufbau der „update“-Methode

Info zu update siehe: https://www.w3schools.com/php/php_mysql_update.asp

Update Data In a MySQL Table Using MySQLi

The UPDATE statement is used to update existing records in a table:

```
UPDATE table_name
SET column1=value, column2=value2,...
WHERE some_column=some_value
```

Beachte:

Die WHERE - Klausel gibt an, welchen Datensatz oder Datensätze aktualisiert werden sollen. Wenn man die WHERE - Klausel weglässt, werden alle Datensätze aktualisiert!

Übung:

- Erstelle eine neue Variable mit dem Namen „\$aenderung“.
Beachte: Starte vor dem „UPDATE“ mit doppelten Anführungszeichen und innerhalb dann mit einfachen Anführungszeichen.

```
29 ▼ if (!$con->set_charset("utf8")){
30     echo "Fehler bei utf8" . $con->error;
31 }
32
33 $aenderung = "UPDATE schueler set vorname = '$vor',
34             nachname = '$nach',
35             geschlecht = '$geschlecht',
36             klasse = '$klasse',
37             gehalt = '$gehalt',
38             geburtsdatum = '$geb' WHERE s_id = '$id';
39             ";
```

- Das alleine reicht nicht, man muss die beiden Methoden „\$con“ und „\$aenderung“ verbinden, damit diese auch durchgeführt werden. Erstelle daher

```
39
40 $eintragen = mysqli_query($con, $aenderung);
41
```

- Darunter eine abschließende Meldung mit einem „echo“:

```
41
42 echo "<h5>Der Schüler wurde erfolgreich bearbeitet!</h5>";
43
```

- Zum Abschluss noch einen Button, der zurück zur Tabelle führt:

```
42 echo "<h5>Der Schüler wurde erfolgreich bearbeitet!</h5>";
43
44 echo'<p><a href="6_datatable_pencil.php"><input class="btn btn-success btn-lg btn-block"
45 type="submit" value="Zurück zur Tabelle"></a></p>';
46 ?>
```

Ergebnis:

