

Lucrarea 3: Dezvoltarea de aplicații PHP cu baze de date MySQL Insert, Update, Delete

1. Obiectivele lucrării

- dezvoltarea de aplicații PHP cu baze de date MySQL: inserarea, ștergerea și modificarea unor înregistrări într-o tabelă a bazei de date .

2. Dezvoltarea de aplicații PHP cu baze de date MySQL

2.1 Adăugarea într-o tabelă a bazei de date

Pentru a putea insera un nou articol într-o tabelă a bazei de date, este nevoie de un formular HTML pentru preluarea de la utilizator a valorilor ce se doresc introduse și transmiterea acestora către scriptul PHP care va realiza inserarea.

În exemplul de mai jos, se poate vedea cum se realizează inserarea unei noi înregistrări într-o tabelă MySQL.

Se consideră baza de date facultate care conține tabelele `studenti_AC`, `note` și `discipline`, cu următoarea structură:

```
CREATE DATABASE facultate;
USE facultate;

CREATE TABLE studenti_AC (Marca char(6) primary key, Nume varchar(30),
Prenume varchar(30), An_Studiu integer);

CREATE TABLE discipline(codDisciplina int primary key, Disciplina varchar(30),
An_Studiu integer);

CREATE TABLE note(Marca char(6) references studenti (Marca), codDisciplina int
references discipline (codDisciplina), Nota double);
```

Tabela `studenti_AC` conține datele personale ale studenților dintr-un anumit an de studiu, de la facultatea de Automatică și Calculatoare, iar tabela `note`, conține notele obținute la disciplinele din anul respectiv. Tabela `studenti_AC` este tabelă părinte, iar tabela `note` este tabelă copil, aceasta din urmă fiind relaționată cu tabela `studenti` prin câmpul `Marca`.

Tabela `discipline` conține denumirea disciplinelor de la fiecare an de studiu, împreună cu codul lor. Tabela `note` (copil) este relaționată cu tabela `discipline` (părinte) prin câmpul `codDisciplina`.

Formularul HTML (din fișierul `introducere_studenti.html`) care permite transmiterea parametrilor `marca`, `numele` și `prenumele` studentului și anul din care face parte este prezentat mai jos:

```
<form method="POST"
action="http://localhost/introducere_studenti.php">
```

```

<table border="3" align="center" BGCOLOR="Silver">
<tr>
<td>Marca:</td>
<td><input type="text" name="marca_form"></td>
</tr>
<tr>
<td>Nume:</td>
<td><input type="text" name="nume_form"></td>
</tr>
<tr>
<td>Prenume:</td>
<td><input type="text" name="prenume_form"></td>
</tr>
<tr>
<td>An studiu:</td>
<td>
<select name="an_form">
<option value="1">An 1</option>
<option value="2">An 2</option>
<option value="3">An 3</option>
<option value="4">An 4</option>
</select>
</td>
</tr>
<tr>
<td colspan="2" align="center">
<input type="SUBMIT" value="Add">
<input type="RESET" value="Reset">
</td>
</tr>
</table>
</form>

```

Acest fișier HTML permite, în cazul de față, transmiterea către scriptul PHP a 4 parametri (*marca*, *nume*, *prenume*, *an* - cu valori introduse de la tastatură) pentru a se putea adăuga un nou articol (complet) în tabela *studenti_AC* a bazei de date.

Marca:	LH2233
Nume:	Tudose
Prenume:	Vlad
An studiu:	An 2 ▼
<input type="button" value="Add"/> <input type="button" value="Reset"/>	

Fig. 2.1. Formularul HTML de adăugare nou articol în tabela *studenti_AC*

Scriptul PHP spre care sunt transmise valorile parametrilor și care va realiza adăugarea propriu-zisă în tabela bazei de date este următorul:

```

<?php
mysql_connect("localhost","root","") or die ("Nu se poate
conecta la serverul MySQL");
mysql_select_db("facultate") or die("Nu se poate selecta baza
de date");

$marca=$_POST['marca_form'];
$nume=$_POST['nume_form'];
$prenume=$_POST['prenume_form'];
$an=$_POST['an_form'];

//se verifica daca studentul care se doreste sa se insereze nu exista deja in baza de date
//daca nu exista, se insereaza; daca exista, se afiseaza un mesaj de eroare

$query=mysql_query(„select count(*) from studenti_AC where marca=' $marca”);
$row=mysql_fetch_row($query);
$nr=$row[0];

if ($nr==0){
// adăugare cu parametri
$query1=mysql_query("insert into studenti_AC
values ( ' $marca' , ' $nume' , ' $prenume' , $an) ”);

//selectarea și afișarea doar a înregistrării adăugate
$query2=mysql_query("select * from studenti_AC where
marca=' $marca' ”);

//calculeaza nr. de inregistrari returnate prin interogare
$nr_inreg=mysql_num_rows($query2);

if ($nr_inreg>0){

echo "<table border='2' align='center'>";
$coln=mysql_num_fields($query2); //nr. de campuri
echo"<tr bgcolor='silver'>";

//realizare automată a capului de tabel (conținând toate câmpurile)
for ($i=0; $i<$coln; $i++){
//numele câmpurilor ca și cap de tabel
$var=mysql_field_name($query2, $i);
echo "<th> $var </th>";
}
echo"</tr>";

//afiseaza inregistrarile gasite in urma interogarii
while($row=mysql_fetch_row($query2)){
echo"<tr>";
foreach ($row as $value){
echo "<td>$value</td>";
}
echo"</tr>";
}
echo"</table>";
}
else{

```

```

echo"<center>";
echo "Nu s-a gasit nici o inregistrare!!!";
echo"</center>";
}
}
else{
echo"<center>";
echo "Studentul respectiv exista deja in baza de date!";
echo"</center>";
}
mysql_close();
?>

```

Rezultatul vizibil pe ecran, produs în urma execuției comenzii SQL de afișare *SELECT*... este prezentat în fig. 3.1. (*Observație*: Comanda *INSERT* nu va genera un rezultat vizibil pe ecran).

Marca	Nume	Prenume	An_Studiu
LH2233	Tudose	Vlad	2

Fig. 2.2. Confirmare adăugare

De asemenea, în locul unei comenzi *mysql_query("SELECT... compacte*, se poate scrie echivalent:

```

//string de interogare memorat într-o variabilă
$com= 'select * from table1';
$query=mysql_query($com); //interogare

```

2.2 Ștergere înregistrare dintr-o tabelă a bazei de date

Ștergerea unor articole dintr-o tabelă a unei baze de date (ștergere presupusă a fi cu o confirmare prealabilă) implică 2 operații:

- realizarea unei interogări a bazei de date (pe baza unor parametri de intrare pentru interogare) și afișarea înregistrărilor găsite, care vor fi șterse;
- confirmarea (sau anularea) ștergerii efective, pe baza informațiilor anterior afișate în urma interogării.

Fișierul HTML (*stergere_Studenti.html*) pentru realizarea unei interogări a tablei în vederea ștergerii (interogare făcută după parametrii *-nume* și *prenume*- introdusi de la tastatură) este prezentat în listingul următor (având rezultatul prezentat în fig. 2.4):

Nume:	<input type="text" value="Tudose"/>
Prenume:	<input type="text" value="Vlad"/>
<input type="button" value="Sterge"/> <input type="button" value="Anulare"/>	

Fig. 2.3 Formular ștergere

```

<form method="POST"
action="http://localhost/stergere Studenti1.php">
<table border="3" align="center" bgcolor="silver">
<tr>
<td>Nume:</td>

```

```

<td>
<input type="text" name="nume_form">
</td>
</tr>
<tr>
<td>Prenume:</td>
<td>
<input type="text" name="prenume_form">
</td>
</tr>
<tr>
<td colspan="2" align="center">
<input type="SUBMIT" value="Sterge" >
<input type="RESET" value="Anulare" >
</td>
</tr>
</table>
</form>

```

Fișierul PHP (*stergere_Studenti1.php*) pentru interogarea prealabilă ștergerii, este prezentat în continuare :

```

<?php

mysql_connect("localhost","root","") or die ("Nu se poate
conecta la serverul MySQL");
mysql_select_db("facultate") or die("Nu se poate selecta baza
de date");

$nume=$_POST['nume_form'];
$prenume=$_POST['prenume_form'];

// căutare după câmpul nume a înregistrărilor care vor fi șterse
$query=mysql_query("select * from studenti_AC where
nume='$nume' and prenume='$prenume'");

//calculeaza nr. de inregistrari returnate prin interogare
$numr_inreg=mysql_num_rows($query);

if ($numr_inreg>0){
    echo "<center>";
    echo "S-au gasit " . $numr_inreg . " inregistrari";
    echo "</center>";

    // creare tabelă de afișare rezultate
    echo "<table border='3' align='center'>";
    //numarare campuri
    $coln=mysql_num_fields($query);
    echo"<tr bgcolor='silver'>";
    //realizare automată a capului de tabel (conținând toate câmpurile)
    for ($i=0; $i<$coln; $i++){
        //numele câmpurilor ca și cap de tabel
        $svar=mysql_field_name($query,$i);
        echo "<th> $svar </th>";
    }
}

```

```

echo"</tr>";
//extragere informații și afișare
while (list ($marca, $nume, $prenume, $an) =
mysql_fetch_row($query)) {
    print (" <tr>".
        " <td>$marca</td>".
        " <td>$nume</td>".
        " <td>$prenume</td>".
        " <td>$an</td>".
        " </tr>");
}
echo"</table>";

// Apelarea scriptului de ștergere efectivă/anulare (cu transmitere mai departe
// a parametrilor de intrare, în cazul de față 'nume' după care se face cautarea)

echo "<form method=' POST'
action=' http://localhost/stergere_Studenti2.php'>";

// "pasare", transmitere mai departe a parametrului nume ($nume)
//sub numele ' nume1'

echo "<input type='hidden' name=' nume1_form'
value=" .$_POST['nume_form'] . ">";
echo "<input type=' SUBMIT' value=' Stergere efectiva'>";
echo "</form>";
// link pt. revenire la pagina de start și anulare ștergere
echo "<a HREF=' http://localhost/stergere_Studenti.html'>
Renunț și revin...</a>";
}
else
    die("Nu gasesc nici o inregistrare ...");
mysql_close();
?>

```

Pentru afișarea rezultatelor obținute în urma interogării, mai sus s-a folosit construcția de limbaj *list()*, care permite asignarea unor valori unei liste de variabile (asemănătoare practic unui șir), printr-o singură operație, valorile asignate fiind cele extrase cu ajutorul funcției *mysql_fetch_row()*.

În fișierul de script PHP anterior prezentat, se realizează practic o simplă interogare a tabelii bazei de date, cu afișarea rezultatelor pe ecran (fig. 2.4).



Fig. 2.4 Confirmare ștergere

Ceea ce conferă posibilitatea luării deciziei de ștergere a articolelor astfel vizualizate se realizează prin bucata de program inclusă în bucla *if*, de verificare a numărului de articole găsite, constând practic într-un formular HTML *<form...>*.

Aceasta este următoarea (PHP trimițând practic spre ecran un script HTML folosind comanda *echo*) :

```
echo „<form method='POST'
action='http://localhost/stergere_Studenti2.php'>”;

// ”pasare”, transmitere mai departe a parametrului nume ($nume)
//sub numele ‘ nume1’

echo „<input type='hidden' name='nume1'
value=" .$_POST['nume_form'] .">”;
echo „<input type='SUBMIT' value='Stergere efectiva'>”;
echo „</form>”;
```

Secvența de cod prezentată, transmite o valoare a parametrului de interogare, memorată în variabila *\$nume* (parametru utilizat aici pentru o simplă afișare cu SELECT), spre un alt parametru (*nume1_form*) care va fi pasat spre fișierul PHP de ștergere efectivă. Se remarcă în acest caz, tipul *type="hidden"* utilizat pentru caseta de transfer a parametrului, care face această casetă invizibilă pentru utilizator.

Fișierul php (*stergere_Studenti2.php*) pentru ștergere efectivă are în acest caz o structură extrem de simplă:

```
<?php
mysql_connect("localhost","root","") or die ("Nu se poate
conecta la serverul MySQL");
mysql_select_db("facultate") or die("Nu se poate selecta baza
de date");
$nume1=$_POST['nume1_form'];
// ștergere efectivă
$query =mysql_query("DELETE FROM studenti AC where
nume=' $nume1 '");
mysql_close ();
?>
```

2.3 Modificarea unei înregistrări dintr-o tabelă a unei baze de date

Modul de modificare a unei înregistrări presupune următoarele:

- o operație de căutare parametrizată a înregistrării care se dorește a fi modificată și o afișare a ei
- modificarea efectivă (folosind afișarea realizată)

Fișierul pentru transmiterea parametrilor de căutare a înregistrării care se dorește a fi modificată (căutarea făcându-se după 2 parametri – vezi fig. 2.5) este următorul:

```
<form method="POST"
action="http://localhost/Update_Studenti1.php">
<table border="3" align="center" bgcolor="Silver">
<tr>
<td>
Nume: <input type="text" name="nume_form">
</td>
</tr>
```

```

<tr>
<td>
Prenume: <input type="text" name="prenume_form">
</td>
</tr>
<tr>
<td colspan="2" align="center">
<input type="SUBMIT" value="Cauta">
<input type="RESET" value="Anulare">
</td>
</tr>
</table>
</form>

```

Fig. 2.5. Căutare pentru modificare (practic o simplă căutare)

Fișierul pentru interogare în vederea modificării (*update_Studenti1.php*) are următorul cod:

```

<?php
mysql connect("localhost","root","") or die ("Nu se poate
conecta la serverul MySQL");
mysql_select_db("facultate") or die("Nu se poate selecta baza
de date");
$nume=$_POST['nume_form'];
$prenume=$_POST['prenume_form'];

// căutarea înregistrării care va fi modificată
$query=mysql_query("select * from studenti_AC where
nume='$nume' and prenume='$prenume'");

$nr_inreg=mysql_num_rows($query);

if ($nr_inreg>0){
echo "<center>";
echo "S-au gasit " . $nr_inreg . " inregistrari";
echo"</center>";

echo "<table border='2' align='center'>";
echo"<tr bgcolor='silver'>";
$coln=mysql_num_fields($query);

for ($i=0; $i<$coln; $i++){
    $var=mysql_field_name($query,$i);
    echo "<th> $var </th>";
}
echo"</tr>";

$nr_val=0; //contor indice array
while ($row = mysql_fetch_row($query)){

```



```

echo"<tr>";
foreach ($row as $value) {
    echo "<td BGCOLOR='Yellow'> $value</td>";
    // memorare într-un șir (array) a datelor din articolul găsit
    $sir[$nr_val]=$value;
    $nr_val++;
}
    echo "</tr>";
}
echo "</table>";

// Rezolvarea este valabilă pentru o singură înregistrare găsită
// Pentru mai multe înregistrări găsite, modificările efectuate se aplică asupra tuturor
    echo '<br><hr>'; // trasarea unei linii

// apel script pentru modificarea efectivă
echo „<form method=' POST'
action='http://localhost/update_Studenti2.php'>”;

// transfer (ascuns) spre script a parametrilor de căutare
echo „<input type='hidden' name='nume2_form'
value=" . $sir[1] . ">”;
echo „<input type='hidden' name='varsta2_form'
value=" . $sir[2] . ">”;

// transfer spre script ca parametrii a datelor care pot fi modificate
echo „<table>”;
echo „<tr>”;
echo „<td>”;
echo „<input type='text' name='marca1_form'
value=" . $sir[0] . ">”;
echo „<input type='text' name='nume1_form'
value=" . $sir[1] . ">”;
echo „<input type='text' name='prenume1_form'
value=" . $sir[2] . ">”;
echo „<input type='text' name='an1_form'
value=" . $sir[3] . ">”;
echo „</td>”;
echo „</tr>”;
echo „<tr>”;
echo „<td>”;
echo „<input type='SUBMIT' value='Modifica!' >”;
echo „</td>”;
echo „</tr>”;
echo „</table>”;
echo „</form>”;

// link de revenire și renunțare la modificare
echo '<a HREF="http://localhost/update_Studenti.html"> Renunț
și revin...</a>';
}
else
    die ("Nu gasesc nici o inregistrare ...");
mysql_close();
?>

```

Rezultatul fișierului script PHP anterior, este prezentat în figura 2.6. Ceea ce aduce nou acest script PHP, pe lângă o simplă interogare (căutare cu SELECT) și afișare a articolelor găsite pe ecran (tabelat), constă într-o afișare a datelor găsite care să permită și o modificare a lor (vezi fig. 2.6). Cum s-a realizat acest lucru?

S-au gasit 1 inregistrari			
Marca	Nume	Prenume	An_Studiu
LH2231	Tudose	Vlad	3

LH2231 Tudose Vlad 3

[Renunt si revin...](#)

Fig. 2.6. Ecran pentru modificare

În primul rând, în bucla *foreach(...)* încuibată în bucla *while(...)*, apare o secvență de genul:

```
// memorare într-un șir (array) a datelor din articolul găsit
    $sir[$nr_val]=$value;
//incrementare contor număr elemente
    $nr_val++;
```

care permite memorarea tuturor elementelor găsite într-un șir (array-ul *\$sir*).

Spre exemplu, în cazul de față, s-a găsit un singur articol cu 4 elemente, acestea memorându-se într-un șir având 4 elemente (cu indicii 0,1,2 și 3).

În al doilea rând, este utilizată secvența următoare:

```
echo „<form method=' POST'
action=' http://localhost/update_Studenti2.php' >“;

// transfer (ascuns) spre script a parametrilor de căutare
echo „<input type='hidden' name=' nume2_form'
value=" . $sir[1] . ">“;
echo „<input type='hidden' name=' prenume2_form'
value=" . $sir[2] . ">“;

// transfer spre script ca parametrii a datelor care pot fi modificate
echo „<table>“;
echo „<tr>“;
echo „<td>“;
echo „<input type='text' name=' marcal_form'
value=" . $sir[0] . ">“;
echo „<input type='text' name=' nume1_form'
value=" . $sir[1] . ">“;
echo „<input type='text' name=' prenume1_form'
value=" . $sir[2] . ">“;
echo „<input type='text' name=' an1_form'
value=" . $sir[3] . ">“;
echo „</td>“;
echo „</tr>“;
echo „<tr>“;
echo „<td>“;
```

```

echo „<input type='SUBMIT' value='Modifica!' >”;
echo „</td>”;
echo „</tr>”;
echo „</table>”;
echo „</form>”;

```

Primele tag-uri `<INPUT type="hidden">` permit transferul spre fișierul script PHP care realizează modificarea efectivă a parametrilor inițiali de interogare (memorați în primele două elemente ale array-ului `$sir`), acest lucru fiind necesar deoarece acești parametri pot fi ‘alterați’ în urma unei eventuale modificări permise în continuare, după cum se va vedea.

Următoarele 4 tag-uri `<INPUT type="text">` permit afișarea (în casete care permit atât operații de citire cât și de scriere) și alterarea (modificarea) acestor 4 parametri, care vor substitui informația existentă în câmpurile articolului localizat (cu parametrii transmiși din primele două `INPUT`-uri invizibile utilizatorului).

Cu alte cuvinte, această secvența transferă spre script-ul PHP de modificare efectivă 6 parametri:

- 2 parametri pentru localizarea articolului de modificat
- alți 4 parametri pentru substituirea informației din cele 4 câmpuri ale articolului deja localizat

Fișierului PHP (`update_Studenti2.php`) care realizează modificarea propriu-zisă, utilizând comanda SQL `UPDATE`, nu îi mai revine decât o sarcină banală:

```

<?php
mysql_connect("localhost","root","") or die ("Nu se poate
conecta la serverul MySQL");
mysql_select_db("facultate") or die("Nu se poate selecta baza
de date");

echo $nume2=$_POST['nume2_form'];
echo $prenume2=$_POST['prenume2_form'];

echo $marcal=$_POST['marcal_form'];
echo $numel=$_POST['numel_form'];
echo $prenume1=$_POST['prenume1_form'];
echo $an1=$_POST['an1_form'];

// modificare efectivă
$query =mysql_query("update studenti_AC set
marca=' $marcal', nume=' $numel',
prenume=' $prenume1', an_studiu=' $an1' where nume=' $nume2' and
prenume=' $prenume2' ");

// afișare mesaj de eroare pentru date incorect introduse (dacă se dorește)
$var=mysql_error();
echo $var;
echo "OK, am modificat!";
mysql_close ();
?>

```

3. Probleme propuse

1. Se vor studia si rula exemplele din aceasta lucrare
2. Se considera baza de date *facultate*, care contine tabelele *studenti*, *discipline* si *note*, a caror structura este prezentata mai sus. Tabela discipline si tabela note se populeaza cu date din PHPmyAdmin.

Se cere crearea unui meniu care sa contina, urmatoarele optiuni: Inserare Studenti, Stergere Studenti, Update Date Studenti), urmatoarele optiuni (sub forma de link-uri):

- afisarea tuturor studentilor din tabela studenti;
- inserarea unui student nou;
- stergerea unui student;
- update date studenti
- afisarea celei mai mici note a unui student pe care il cautam dupa marca (afisam intr-un tabel marca, numele, prenumele, anul de studiu, disciplina si nota);
- afisarea numarului de studenti care au promovat o anumita disciplina (afisam disciplina, anul si numarul de studenti);
- afisarea unui anumit student, cu toate disciplinele si notele obtinute (afisam intr-un tabel marca, numele, prenumele, anul de studiu, disciplina si nota), iar sub tabel media notelor pe anul respectiv;
- actualizarea notei la o anumita disciplina, a unui student. Se cauta un student dupa marca si disciplina in tabela note. Se afiseaza studentul cu toate datele sale din tabela note, de pe linia care indeplineste conditiile de cautare, apoi intr-un form se introduce noua nota si se face update in tabela.
- stergerea din tabela note a unei discipline pentru un anumit student. Se creeaza un form in care se introduce marca unui student si disciplina. Datele respective se transmit catre un script PHP care va face stergerea inregistrarii respective.

Din link-urile din meniu se transmit parametri cu metoda GET, pentru fiecare optiune in parte. In scriptul PHP se preiau acesti parametri si cu ajutorul unui *if* se verifica ce optiune din meniu s-a selectat. Pe fiecare ramura se include fisierul cu rezolvarea cerintei corespunzatoare. Se foloseste *include*.

Exemplu de utilizare:

```
<a href=http://localhost/prelucrare.php?opt=1">Inserare
Studenti </a>
<a href=http://localhost/prelucrare.php?opt=2">Update Date
Studenti </a>
<a href=http://localhost/prelucrare.php?opt=3">Stergere
Studenti </a>
```

In fisierul prelucrare.php:

```
$o=$ GET[, opt' ];
if ($o==1){
include („http://localhost/inserare.php”);
}
else if ($o==2){
include („http://localhost/updatate.php”);
}
else
```

```
include („http://localhost/stergere.php“);  
}
```