

Baze de Date

Cap. 7. Dezvoltare aplicații BD folosind MySQL și PHP



Textbook: Ramakrishnan, Gehrke, "Database Management Systems", McGraw Hill, C15

2023 UPT

Conf.Dr. Dan Pescaru

SGBD MySQL

1. MySQL - sistem de gestiune a bazelor de date relaționale tip open-source
2. A fost creat de compania suedeză MySQL AB (1995) și distribuit sub licență GNU GPL
3. Este des utilizat împreună cu limbajul PHP pentru baze de date Web mici și mijlocii
4. Este inclus în pachetele LAMP și WAMP



Despre MySQL

1. Include unelte de gestionare și configurare precum MySQL Administrator, MySQL Query Browser și phpMyAdmin
2. Rulează sub Linux, Windows, MacOS...
3. Cumpărat și dezvoltat de Oracle (v.5-8)
4. Derivat în MariaDB (Michael Widenius)
5. Versiuni MySQL Community Server (gratuit) și Enterprise Server (comercial)

Limbajul HTML

1. HTML = Hyper Text Mark-up Language

2. Este un limbaj de marcare (formatare)

- Folosit pentru aranjarea informațiilor în paginile web
- Se bazează pe tag-uri HTML (incluse în <...>)
- Este interpretat de navigatoare (browsere) Web



3. Este un standard întreținut de W3C



4. Prin intermediul său (+ limbaje adiționale precum CSS sau SVG) se pot aranja și formata într-o pagină Web diverse elemente precum text, tabele, imagini

5. Organizează pagina Web în două secțiuni

- HEAD – conține meta-informații despre pagină
- BODY – conține informația propriu-zisă

Exemplu pagină HTML

```
<head>
  <title>Prima pagina HTML</title>
  <style>
    table, th, td { border:2px solid blue; border-collapse:collapse; }
  </style>
</head>
<body>
  <h1>Pagina HTML exemplu</h1>
  <p>
    Aceasta pagina este un exemplu de utilizare taguri HTML.
  </p>
  <h2>Text formatat</h2>
  <b>Acesta</b> este un <i>text</i> formatat folosind <u>>taguri <b>HTML</b></u>. <br>
  Se pot folosi diverse <span style="color:blue">culori</span> si <span style="font-family:'Courier New'; font-size:15px">fonturi</span>
  <h2>Imagini</h2>
  Acesta este o imagine adaugata folosind tagul . <br>
  <h2>Liste</h2>
  Listele pot fi ne-numerotate sau numerotate (ordonate).<br>
  Lista ne-numerotata
  <ul>
    <li>Prima intrare</li>
    <li>A doua intrare</li>
  </ul>
  <br>
  Lista numerotata
  <ol>
    <li>Prima intrare</li>
    <li>A doua intrare</li>
  </ol>
  <br>
  <h2>Tablouri</h2>
  Tablourile HTML contin randuri si coloane definite cu <tr> si respectiv <th> si <td> pentru cap de colana si <td> pentru continut
  Tablou <b>persoane</b>
  <table>
    <tr style="background-color: #D6EEEE;"> <th>Nr</th><th>nume</th><th>varsta</th> </tr> <!-- cap de tabel -->
    <tr> <td>1</td><td>Popescu</td><td>23</td> </tr>
    <tr> <td>2</td><td>Ionescu</td><td>27</td> </tr>
    <tr> <td>3</td><td>Vasilescu</td><td>22</td> </tr>
  </table>
</body>
```

Exemplu pagină HTML

pagina.html

Pagina HTML exemplu

Aceasta pagina este un exemplu de utilizare taguri HTML.

Text formatat

Acesta este un *text* formatat folosind taguri HTML.
Se pot folosi diverse **culori** si fonturi

Imagini

Acesta este o imagine adaugata folosind tagul .



Liste

Listele pot fi ne-numerate sau numerotate (ordonate).

Lista ne-numerotata

- Prima intrare
- A doua intrare

Lista numerotata

1. Prima intrare
2. A doua intrare

Tablouri

Tablourile HTML contin randuri si coloane definite cu <tr> si respectiv <th> pentru cap de colana si <td> pentru continut de coloana.

Tablou **persoane**

Nr	nume	varsta
1	Popescu	23
2	Ionescu	27
3	Vasilescu	22

Limbajul PHP

1. Limbaj de tip scripting, open-source, rulează pe partea de server
 - Folosit pentru a genera pagini web dinamice
 - Scripturi PHP încorporate în paginile HTML
2. Lansat în 1995, de Rasmus Lerdorf ca un set de instrumente de script Perl/CGI (Personal Home Page sau PHP)
3. PHP2 lansat în 1997 (ca "PHP Hypertext Processor")
4. PHP5 lansat în 2004 (> 100 biblioteci, > 1.000 de funcții, programare OO extinsă etc.)
5. PHP8 lansat în 2020 (compilare just-in-time, expresii regulate, operator Nullsafe etc.)



Avantaje PHP

1. Limbajul interpretat, scripturile sunt analizate în timpul execuției și nu compilate în prealabil
2. Execut pe partea de server (ascuns pentru browserul client – opțiunea Vizualizare sursă în browser nu afișează scriptul)
3. Diverse funcții încorporate permit aplicații variate - grafică, e-mail, pdf etc.
4. Compatibil cu multe SGBD-uri populare (MySQL, Oracle, MsSQL Server, IBM DB2...)

Codul PHP

1. Similar ca sintaxa/instrucțiuni de bază cu C/C++/Java
2. Inclus în fișiere HTML folosind tag-ul `<?php ... ?>`. Se permit mai multe secțiuni PHP în aceeași pagină
3. Permite programare structurată / POO
4. Instrucțiunile se despart prin `;`
5. Permite comentarii gen C/C++/Shell (`/* ... */`, `// ...`, `# ...`)

Ref: <https://www.w3schools.com/php/>

Variabilele PHP

1. Numele unei variabile începe cu \$
2. Sunt case sensitive (\$a != \$A)
3. PHP este un limbaj slab tipizat
4. Variabilele au vizibilitate locală sau globală
 - Cele locale restricționate în cadrul unei funcții
 - Cele globale utilizabile și în afara funcțiilor
5. Există câteva variabile predefinite
 - Pt. date din formulare (\$_GET, \$_POST, \$_FILE)
 - Date aplicație Web (\$_SESSION, \$_COOKIE)
 - Date despre serverul Web (\$_SERVER)

Tipurile de date PHP

1. Tipuri de bază

- Logic (boolean) `true, false`
- Întreg (integer) `[-2,147,483,648 .. 2,147,483,647]`
- Real (float)
- Șir de caractere (string)
- Tablou, tablou asociativ (array)

2. Tipuri complexe

- Obiecte PHP (object)
- Tipuri specifice (pdfdoc, pdfinfo)

3. Lipsă valoare – `null`

Generare HTML

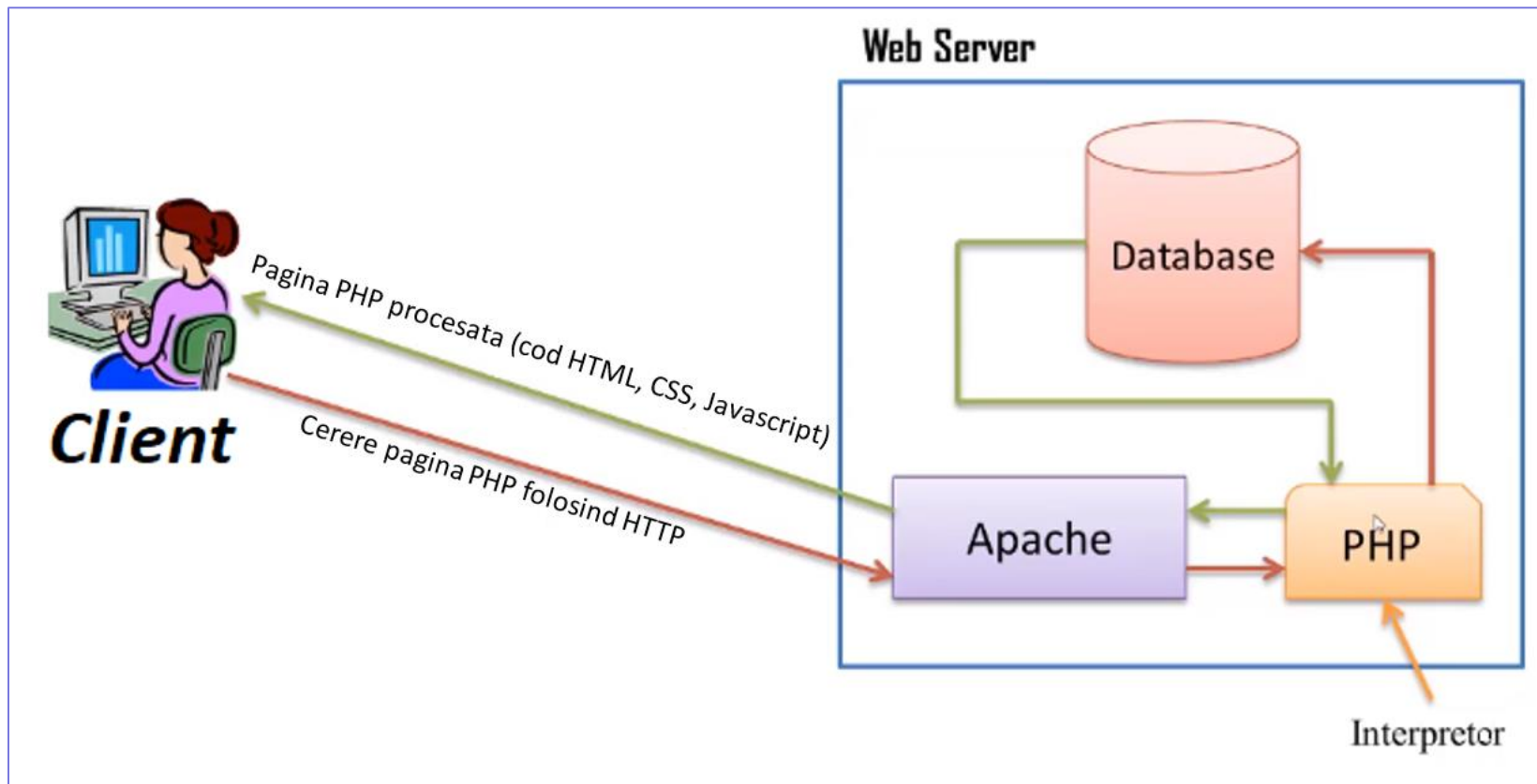
1. Prin instrucțiunea PHP "echo" sau prin unele funcții PHP (print, printf, print_r, die, ...).
2. Se va insera informații în pagina încărcată de browser-ul web al clientului

<?php

```
$a = 3;                // Variabilă numerică
$b = "abc";            // Variabilă șir
echo $a;               // Inserează: 3
echo $a, $b;           // Inserează: 3abc
echo "1+2=".$b;        // Inserează: 1+2=3
echo "1+2=$b";         // Inserează: 1+2=3
echo '1+2=$b'. '<br />'; // Inserează: 1+2=$b + rând nou
```

?>

Interpretare limbaj PHP



Preluata dupa: <https://www.devopsschool.com/>

Operatori PHP

1. Aritmetici

- (+, -, *, /, %, ++, --)

2. Concatenare șiruri, evaluare șiruri

- (.), ("...")

3. Comparații

- (==, ===, !=, <>, <, <=, >, >=)

4. Pentru tipuri de date

- (instanceof, is_bool, is_int, is_string, is_object ...)

5. Atribuire

- (=, +=, -=, *=, /=, %=, .=)

6. Extragere referință

- (&\$var)

Structuri condiționale de control

1. IF/ELSE/ELSEIF

```
<?php
    if($a<3) {
        echo $a;
        $a=5;
    }
    elseif($a<6) {
        echo "corect!";
    }
    else
        echo "greșit";
?>
```

2. SWITCH

```
<?php
    switch ($a) {
        case 0:
            echo "Este zero";
            break;
        case 1:
            echo "Este unu";
            break;
        default:
            echo "Prea mare";
    }
?>
```

Cicluri WHILE

1. WHILE

```
<?php
    $a=0;
    while ($a<5)
    {
        echo "a = $a; ";
        $a ++;
    }
?>
```

2. DO WHILE

```
<?php
    $a = 0;
    do {
        echo "a = $a; ";
        $a++;
    } while ($a < 5);
    echo "<br />done.";
?>
```

Cicluri FOR

1. FOR

```
<?php
for ($a=1; $a<$b; $a++)
{
    if ($a >5) {
        break;
    }
    echo $a;
}
?>
```

2. FOREACH

```
<?php
$bon = array(
    "pâine" => 4.5,
    "roșii"  => 9.0,
    "carne"  => 26.5
);
foreach ($bon as
    $prod => $pret) {
    echo "Prețul pt. $prod
        este $pret lei.";
}
?>
```

Accesare date formulare HTML

1. Pentru formularul (fișier "citire.html")

```
<form name="formular" method="post" action="frmproc.php">
  <p>Date personale <br />
  Nume<input type="text" name="nm" maxlength="36" />
  <br />
  Prenume<input type="text" name="pn" maxlength="36" />
  <br /> Transport favorit
  <select name="transp">
    <option value="Masina">Masina</option>
    <option value="Tren">Tren</option>
    <option value="Avion">Avion</option>
  </select> <br />
  <input type="submit" name="send" value="Salvare" />
  <input type="reset" name="del" value="Resetare" />
</form>
```


Cod accesare date formulare HTML

1. Fișierul PHP țintă "frmproc.php"

<h1>Date transmise:**</h1>**

<?php

\$nume = **\$_POST**["nm"];

\$prenume = **\$_POST**["pn"];

\$transport = **\$_POST**["transp"];

echo "Nume: ".\$nume."**
**";

echo "Prenume: ".\$prenume."**
**";

echo "Opțiune transport: ".\$transport."**
**";

?>

Funcții PHP

1. Funcțiile trebuie definite înainte de a putea fi apelate
2. Antetul unei funcții este de forma **function** numeFct(\$arg1, \$arg2, ..., \$argN)
3. Nu este specificat niciun tip returnat
4. Spre deosebire de variabile, la numele funcțiilor nu se face distincție între caractere mici/mari (fct1(...), FCT1(...) sau Fct1(...))
5. Pot fi grupate în biblioteci utilizate cu funcția PHP **include**("numeBibliotecă.php")

Funcții PHP de acces la BD MySQL

1. `new mysqli($host, $user, $pass, $dbName)`
2. `mysqli_query()` // trimite o interogare
3. `mysqli_num_rows ()` // valabilă pt. SELECT
4. `mysqli_affected_rows ()` // pt. INSERT, UPDATE, DELETE
5. `mysqli_fetch_array()` // returnează un tablou
6. `mysqli_fetch_assoc()` // returnează un tablou asociativ
7. `mysqli_fetch_object()` // returnează un obiect
8. `mysqli_error()` // returnează ultima eroare
9. `mysqli_close()` // închide conexiunea la server

Exemplu de acces la o BD mySQL

```
<?php
```

```
$host = 'localhost';
```

```
$username = 'utilizator';
```

```
$passwd = '12345';
```

```
$dbName = 'persoane';
```

```
$conn = mysqli_connect($host, $username, $passwd,  
                        $dbName);
```

```
if ($conn->connect_error){
```

```
    die('Nu ma pot conecta la server: `.
```

```
        mysqli_error($conn));
```

```
}
```

```
?>
```

Citirea datelor dintr-o BD mySQL

```
<?php
$query = "SELECT nume, prenume, transp FROM Persoana";
$rez = mysqli_query($conn, $query);
if (!$rez) die(' Interogare eşuată: '. mysqli_error($conn));
$nr_inreg = mysqli_num_rows($rez);
echo "<h3>".$nr_inreg." persoane in BD.</h3>";
echo "<table><tr><th>Nume</th>";
echo "<th>Prenume</th><th>Transport</th></tr>";
while($inreg = mysqli_fetch_assoc($rez)) {
    echo "<tr><td>".$inreg["nume"]."</td>";
    echo "<td>".$inreg["prenume"]."</td>";
    echo "<td>".$inreg["transp"]."</td></tr>";
}
echo "</table>";
?>
```