

Databases

Cap. 11. Web databases. PHP+MySQL



Textbook: Ramakrishnan, Gehrke, "Database Management Systems", McGraw Hill, 2003

2017 UPT

Conf.Dr. Dan Pescaru

About PHP

1. Open-source, server-side scripting language
 - Used to generate dynamic web-pages
 - PHP scripts embedded within HTML pages
2. It began in 1995 when Rasmus Lerdorf developed a Perl/CGI script toolset (Personal Home Page or PHP)
3. PHP 2 released 1997 (PHP now stands for Hypertext Processor)
4. PHP5 released 2004 (113 lib, >1.000 functions, extensive OO programming)

PHP Characteristics

1. Interpreted language, scripts are parsed at run-time rather than compiled beforehand
2. Executed on the server-side (hidden for client browser – View Source does not display it)
3. Various built-in functions allow for fast development (graphics, mail, pdf etc.)
4. Compatible with many popular databases (MySQL, Oracle, IBM DB2 ...)

PHP code

1. Structurally similar to C/C++/Java
2. PHP script must be enclosed in the reserved PHP tag '`<?php ... ?>`' (more than one script/page is allowed)
3. Supports procedural and object-oriented paradigm
4. All PHP statements end with a semi-colon
5. Standard C, C++, and shell comments (`/* ... */`, `// ...`, `# ...`)

PHP Variables

1. All begin with a "\$".
2. They are case-sensitive (\$a != \$A)
3. PHP is dynamic, loosely typed language
4. Global and locally-scoped variables (global variables can be used anywhere, local variables restricted to a function or class)
5. Several predefined variables (system variables)
 - Forms data exchange: \$_POST, \$_GET, \$_FILES
 - App data: \$_SESSION, \$_COOKIE
 - Server info: \$_SERVER

PHP Data types

1. Basic types

- integer – integer numbers
- double – real numbers
- string – character strings
- array – heterogeneous arrays, ordered maps (associative arrays)

2. Complex types

- object – PHP objects
- pdfoc, pdfinfo - for PDF generation

Dynamic web content

1. Through PHP 'echo' command or some PHP functions (print, printf, print_r, die, ...). Typical usage is to send data to the client's web-browser

```
<?php
```

```
$a = 3;           // Numerical variable
$b = "abc";       // String variable
echo $a;          // Outputs: 3
echo $a,$b;       // Outputs: 3abc
echo "1+2=".$b;   // Outputs: 1+2=3
echo "1+2=$b";    // Outputs: 1+2=3
echo '1+2=$b'.<br />; // Outputs: 1+2=$b and new line
```

```
?>
```

PHP operators

1. Arithmetic (+, -, *, /, %, ++, --)
2. String concatenation (.)
3. Comparison (==, ===, !=, <>, <, <=, >, >=)
4. Type (instanceof, is_bool, is_int, is_string, is_object ...)
5. Assignment (=, +=, -=, *=, /=, %=, .=)
6. Reference (&\$var)

Conditional control structures

1. IF/ELSE/ELSEIF

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

2. SWITCH

```
<?php
    switch ($a) {
        case 0:
            echo "Is zero";
            break;
        case 1:
            echo "Is one";
            break;
        default:
            echo "To big";
    }
?>
```

While loop structures

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.";
?>
```

For loop structures

1. FOR

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

2. FOREACH

```
<?php
$offer = array(
    "bread" => 4.5,
    "melon" => 6.0,
    "meat" => 24.5
);
foreach ($offer as $prod =>
    $price) {
    echo "The price for $prod is
    $price lei.";
}
?>
```

Accessing HTML FORM data (I)

1. E.g. – the HTML form:

```
<form name="theForm" method="post" action="frmproc.php">
  <p>Personal data <br />
  Name<input type="text" name="nm" maxlength="36" />
  <br />
  Surname<input type="text" name="sn" maxlength="36" />
  <br /> Favorite transportation
  <select name="transp">
    <option value="Car">Car</option>
    <option value="Train">Train</option>
    <option value="Airplane">Airplane</option>
  </select> <br />
  <input type="submit" name="send" value="Register" />
  <input type="reset" name="del" value="Reset" />
</form>
```

Accessing HTML FORM data (II)

1. E.g. – the PHP action file (frmproc.php):

<H1>From processing:</H1>

<?php

\$name = \$_POST["nm"];

\$surname = \$_POST["sn"];

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

echo "Name: ".\$name."
";

echo "Surname: ".\$surname."
";

echo "Transportation option: ".\$transport."
";

?>

PHP Function

1. Functions must be defined before then can be called
2. Function headers are of the format
`function functionName($arg_1, $arg_2, ..., $arg_n)`
3. No return type is specified
4. Unlike variables, function names are not case sensitive (fct1(...) or FCT1(...) or Fct1(...))
5. Can be grouped in libraries used with PHP
function `include("libname.php")`

The PHP MySQL library

1. `mysql_connect()`
2. `mysql_select_db()`
3. `mysql_query()` // send a query
4. `mysql_num_rows()` // for SELECT
5. `mysql_affected_rows()` // for INSERT, UPDATE, DELETE
6. `mysql_fetch_row()` // return a numbered array
7. `mysql_fetch_array()` // return an associative array
8. `mysql_fetch_object()` // return an object
9. `mysql_error()`
10. `mysql_close()`

Connecting to MySQL server

<?php

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

```
$username = 'john';
```

```
$pswd = '12345';
```

```
$dbName = 'persons';
```

```
$con = mysql_connect($host, $username, $pswd);
```

```
if (!$con){
```

```
    die('Could not connect to server: '. mysql_error());
```

```
}
```

```
$db = @mysql_select_db($dbName, $con) or
```

```
    die(mysql_error());
```

?>

Retrieving data from MySQL

```
$query = "SELECT name, surname, transp FROM Persons";  
$result = mysql_query($query);  
if (!$result)  
    die(' Query failed: '. mysql_error());  
$num_rows = mysql_num_rows($result);  
echo "<h3>".$num_rows." persons in DB.</h3>";  
echo "<table><tr><th>Name</th>";  
echo "<th>SName</th><th>Transp</th></tr>";  
while($row = mysql_fetch_array($result)) {  
    echo "<tr><td>".$row["name"]."</td>";  
    echo "<td>".$row["surname"]."</td>";  
    echo "<td>".$row["transp"]."</td>";  
}  
echo "</table>";
```