

LECTURE05

PHP-2

CS418/518: WEB PROGRAMMING

BY DR. JIAN WU

Courtesy: presentation slides from Dr. Justin Brunelle

PROBLEM 1: IF-ELSE AND LOOP

- Use PHP you learnt in class (including previous classes) to make a Lottery Page
 - lottery.php
- Game rules:
 - Generate a random integer x in $[1,3]$ as the lucky number
 - Generate a random integer y in $[1,3]$ as your number
 - If y is equal to x , print “you win the same” y times in different headings; otherwise, print “you lose the game” y times in different headings

lottery.php

```
<?php
echo "<h1>Lottery Game</h1>";
$x=rand(1,3);
$y=rand(1,3);
echo "<p>Today's lucky number is <span style=color:red;font-size:200%>$x</span></p>";
echo "<p>Your number is <span style=color:green;font-size:200%>$y</span></p>";
if($y==$x) for($i=1;$i<=$y;$i++) echo "<h$i style=color:#808000;>Congratulations, you win!</h$i>";
else for($i=1;$i<=$y;$i++) echo "<h$i style=color:#800000;>Sorry, you lost :( Try again!</h$i>";
?>
```

- Use `rand(1, 3)` to generate a random number in [1,3]
- Use two variables, `$x` and `$y`, to store random numbers
- Use IF-ELSE to check whether `$y` and `$x` are equal
- Use FOR loop to echo the results `$y` times
- Use color table <http://htmlcolorcodes.com/> to find the color code of olive and maroon, expressed by either RGB or HEX

PROBLEM 2: A LOTTERY FORM

- Make PHP page
 - [lottery3.php](#)
- Rules:
 1. Store information firstname, lastname, and major **in three arrays**
 2. Show all the 5 students' information in a table
 3. Randomly draw a student as the winner

EFFECT

Heading 1 <h1>

Lottery Game 2

Lastname	Firstname	Major
Smith	John	IST
Kim	Anthony	CSE
Davis	Richard	EE
Miller	Thomas	PHYS
Taylor	Jessica	ACCT

Today's Winner is: Anthony Kim

Font color red, size 200%

```
<?php  
echo "<h1>Lottery Game 2</h1>";
```

lottery3.php

```
/* initialization */  
$lastname=array('Smith','Kim','Davis','Miller','Taylor');  
$firstname=array('John','Anthony','Richard','Thomas','Jessica');  
$major=array('IST','CSE','EE','PHYS','ACCT');  
  
/* create table */  
echo "<table border=1>";  
echo "<tr>";  
echo "<th>LastName</th>";  
echo "<th>FirstName</th>";  
echo "<th>Major</th>";  
echo "</tr>";  
for($i=0;$i<sizeof($lastname);$i++){  
    echo "<tr>";  
    echo "<td>$lastname[$i]</td>";  
    echo "<td>$firstname[$i]</td>";  
    echo "<td>$major[$i]</td>";  
    echo "</tr>";  
}  
echo "</table>";  
  
/* output results -*/  
$v=rand(0,4);  
echo "<p><span style=font-size:200%;color:red>Today's winner is $firstname[$v] $lastname[$v]</span></p>";  
?>
```

- Display the students' information in a table using FOR loop
 - use echo to output the table to HTML
- Use “\$v = rand(0,4)” to generate a random number in [0,4] and store it in \$v

PROBLEM 3: A LOTTERY FORM WITH USER INPUT

[lotteryForm.php](#)

```
<html>
    <form method ="post" >
        Pick a student (1-5)!
        <input type="text" name="studentID" /> <br>
        <input type="submit" value = "go"/>
    </form>
    <?php
        $lastname = array ('Smith', 'Kim', 'Davis', 'Miller', 'Taylor');
        $firstname = array('John', 'Anthony', 'Richard', 'Thomas', 'Jessica');
        if (!empty($_POST['studentID'])){
            $v = $_POST['studentID'];
            echo "Today's Winner is: ".$firstname[$v-1]." ".$lastname[$v-1];
        }
    ?>
</html>
```

Pick a student (1-5)!

go

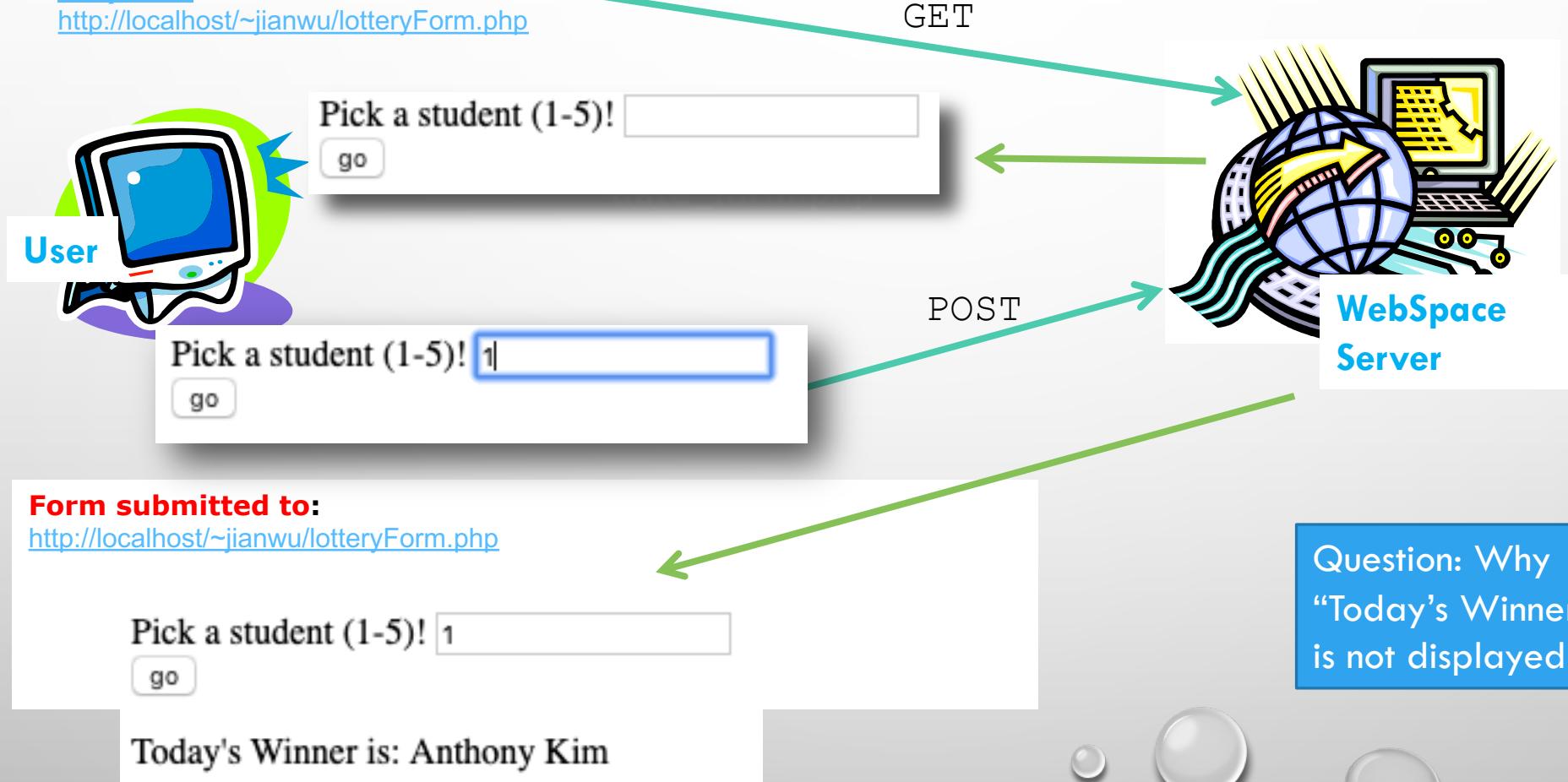
Today's Winner is: Richard Davis

form is an HTML element!
array is a PHP data structure!
\$_POST is used to connect them!

How it works

Request:

<http://localhost/~jianwu/lotteryForm.php>



A more general case

- Input form in one page; PHP response in another page

[lottery4.php](#)

Pick a student (1-5)!

lottery4.php

Today's Winner is: Anthony Kim

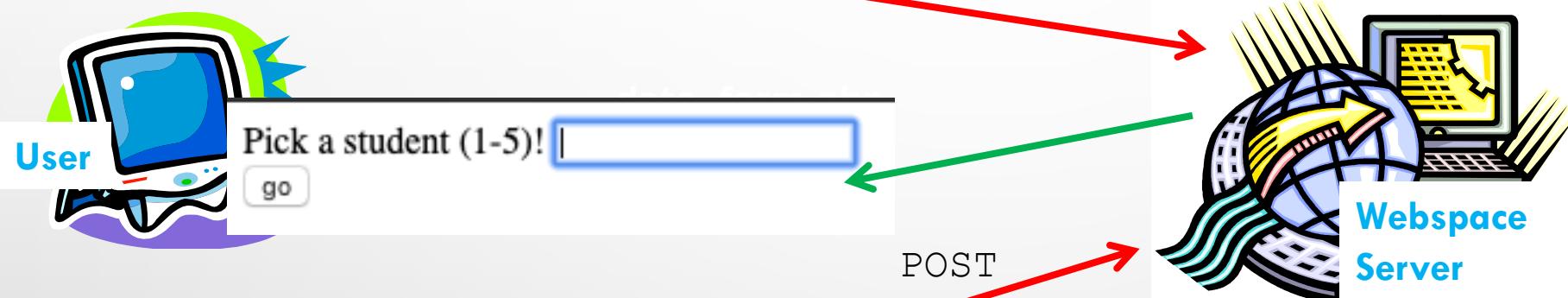
process_lottery4.php

How it works

Request:

<http://localhost/~jianwu/lottery4.php>

GET



Form submission to:

http://localhost/~jianwu/process_lottery4.php

Today's Winner is: Anthony Kim



lotteryForm.php

```
<html>
    <form method = "post" >
        Pick a student (1-5)!
        <input type="text" name="studentID" /> <br>
        <input type="submit" value = "go"/>
    </form>
    <?php
        $lastname = array ('Smith', 'Kim', 'Davis', 'Miller', 'Taylor');
        $firstname = array('John', 'Anthony', 'Richard', 'Thomas', 'Jessica');
        if (!empty($_POST['studentID'])){
            $v = $_POST['studentID'];
            echo "Today's Winner is: ".$firstname[$v-1]." ".$lastname[$v-1];
        }
    ?>
</html>
```

Pick a student (1-5)!

go

Today's Winner is: Anthony Kim

Separate into a **form** page and a **process** page

Pick a student (1-5)!

go

Today's Winner is: Anthony Kim

lottery4.php

```
<html>
    <form action="process_lottery4.php" method ="post" >
        Pick a student (1-5)!
        <input type="text" name="studentID" /> <br>
        <input type="submit" value = "go"/>
    </form>
</html>
```

process_lottery4.php

```
<?php
    $lastname = array ('Smith', 'Kim', 'Davis', 'Miller', 'Taylor');
    $firstname = array('John', 'Anthony', 'Richard', 'Thomas', 'Jessica');
    $v = $_POST['studentID'];
    echo "Today's Winner is: ".$firstname[$v-1]." ".$lastname[$v-1];
?>
```

PROBLEM 4: USER COMMENTS

- Use HTML forms so that visitors can add a comment about your web page
- An HTML form can be one or multiple of the following elements:
 - A text field: single line
 - A textarea field: multiple lines
 - A set of radio buttons: single choice.
 - A dropdown menu: single choice
 - A submit button: submit
 - More: https://www.w3schools.com/php/php_form_complete.asp
- Problem: Write a PHP file to display all data a visitor inputs in a new page after the submit button is clicked

comment.php

```
<html>
<form action="process_comment.php" method="post">
  <table border=0>
    <tr>
      <td> Name: </td>
      <td> <input type = "text" name="username" value = ""/> </td>
    </tr>
    <tr>
      <td> Do you like this page? </td>
      <td> <input type="radio" name="like" value="TRUE"> Yes <input type="radio" name="like" value="FALSE"> No </td>
    </tr>
    <tr>
      <td> Comment: </td>
      <td> <textarea rows="5" cols="80" name="comment"> </textarea> </td>
    </tr>
    <tr>
      <td> Rating: </td>
      <td>
        <select name="rating">
          <option value="1"> 1 </option>
          <option value="2"> 2 </option>
          <option value="3"> 3 </option>
          <option value="4"> 4 </option>
          <option value="5"> 5 </option>
        </select>
      </td>
    </tr>
  </table>
  <input type="submit" value = "go" />
</form>
</html>
```

process_comment.php

```
<?php
$userName=$_POST["username"];
$like=$_POST["like"];
$comment=$_POST["comment"];
$rating=$_POST["rating"];
echo "Hello <b>$userName</b>!<br>";
if($like == "TRUE") {
    echo "I am happy that you <strong>like</strong> this page :) <br/>";
} else {
    echo "I am sorry that you <strong>do not like</strong> this page :( <br/>";
}
echo "Thanks for your comment <strong>[$comment]</strong> <br/>";
echo "You rate my page as Rating <strong>$rating</strong>! ";
?>
```

- This file accepts and processes the input from comment.php
- Pay attention to how variables are passed from HTML forms to the PHP variables
- “TRUE” is a string, not a Boolean.

INTERACT WITH A DATABASE

- Test if database connection is working. In MySQL:
 1. Create a database called php (you do not have to create a table yet).
 2. Create a PHP file like below called `dbtest.php`.

```
<?php  
    $server = "localhost";  
    $sqlUsername = "root";  
    $sqlPassword = "september2019";  
    $databaseName = "php";  
  
    $conn = new mysqli($server, $sqlUsername, $sqlPassword);  
  
    // check connection  
    if ($conn->connect_error) {  
        die("Connection failed: " . $conn->connect_error);  
    }  
    echo "Connected successfully";  
?>
```

MYSQL AUTHENTICATION ERROR

- If you see this error message below:

Warning: mysqli_connect(): The server requested authentication method unknown to the client [caching_sha2_password]

- This is because of the authentication plugin MySQL 8.
- MySQL 8's default plugin is `auth_socket`. Applications will most times expect to log in to your database using a **password**.
- To work around this problem, run the following command in MySQL

```
ALTER USER 'myadmin'@'localhost' IDENTIFIED WITH mysql_native_password by 'yourpassword';
```

TESTING CONNECTION TO THE DATABASE

- Test using the code `dbtest.php`
- Visit [dbtest.php](#) and see if you see “Connected successfully”
- Next: login to the database server and create a database

```
mysql> CREATE DATABASE xschool default character set 'utf8mb4';
mysql> USE xschool;
```

POPULATE YOUR DATABASE

- Create a database called: xschool, using utf8 as the encoding
- Then import the .sql files in the class folder to create tables and insert data.
 - `source MySQL-Create-Tables.sql`
 - `source MySQL-Insert-Data.sql`

These commands assume .sql files are saved under the same directory you currently are.
If .sql files are saved in different directories, you should specify their paths.
- MySQL-Create-Tables.sql creates four tables under xschool database.
 - `show tables;`
- MySQL-Insert-Data.sql inserts data to each table.
 - `desc EMPLOYEE;`
 - `desc DEPARTMENT;`
 - `desc PROJECT;`
 - `desc ASSIGNMENT;`
- Now you have a database of school employees. To view the first 5 entries in the EMPLOYEE table:
`SELECT * FROM EMPLOYEE LIMIT 5;`

PROBLEM 5: SHOW TABLE CONTENT

- Source code: [query.php](#)
- Effect: [query.php](#)

This usually indicates that the username or the password is incorrect.

Query

Which table to query:

Submit

ERROR: database parameters are not correct

PROBLEM 6: USER LOGIN

- Source codes: inside the `login/` folder
 - `login.php`
 - `authentication.php`
 - `logout.php`
 - `main.php`
 - `profile.php`
 - `signup.php`
- Effect: [login.php](#)
- Challenges:
 - Session control:
 - How do pages after the login page know you have logged in?
 - Authentication
 - Can you directly visit the main page without logging in?

Create another database called `php`.

Create a table called `userprofile`, using the code below:

```
CREATE TABLE userprofile(  
    userid varchar(50) NOT NULL,  
    firstname varchar(50) DEFAULT NULL,  
    lastname varchar(50) DEFAULT NULL,  
    email varchar(50) DEFAULT NULL,  
    password varchar(1024) DEFAULT NULL,  
    CONSTRAINT PK_userprofile PRIMARY KEY (userid)  
) ;
```

AUTHENTICATION

[login.php](#)

```
// Authenticate the user
if (authenticateUser($connection, $loginUserId, $loginPassword))
{
    //the user id and password match,
    // set the session
    $_SESSION['db_is_logged_in'] = true;
    $_SESSION['userID'] = $loginUserId;

    // after login we move to the main page
    header('Location: main.php');
    exit;
} else {
    $errorMessage = 'Sorry, wrong username / password';
}
```

[main.php](#)

```
//is the one accessing this page logged in or not?
if (!isset($_SESSION['db_is_logged_in'])
    || $_SESSION['db_is_logged_in'] != true) {
    // not logged in, move to login page
    header('Location: login.php');
    exit;
} else {
    //logged in, display appropriate information
    echo "Hello ", $_SESSION['userID'], "!";
}
```

KEY ISSUES IN SESSION CONTROL

- Use `session_start()` in **every php file** that needs access to session **variables**
 - This function registers this session with the web server and gets a UID for the session
 - This function initializes the `$_SESSION` array to store data
- Session variables are stored in an array named as `$_SESSION`
 - You can specify array index and values as `$_SESSION['variable_name'] = value`
- Session variables are **super global (w.r.t. local and global)**
 - Session variables can last forever, unless
 - You delete them, or system sets a life span for them
- Close the session after all business is finished
 - `session_destroy();`

More about session management in PHP:

<https://www.php.net/manual/en/features.session.security.management.php>

<https://www.php.net/manual/en/book.session.php>

profile.php

must be done first

```
<?php  
    //include information required to access database  
    require 'authentication.php';
```

```
    //start a session  
    session_start();
```

must check both

```
    //still logged in?  
    if (!isset($_SESSION['db_is_logged_in'])  
        || $_SESSION['db_is_logged_in'] != true) {  
        //not logged in, move to login page  
        header('Location: login.php');  
        exit;  
    } else {
```

redirect to another page

display profile information only if logged in



BACKUP SLIDES BEYOND THIS POINT

JIAN WU