

Web Technologies (ICDT6004) – Lecture 10

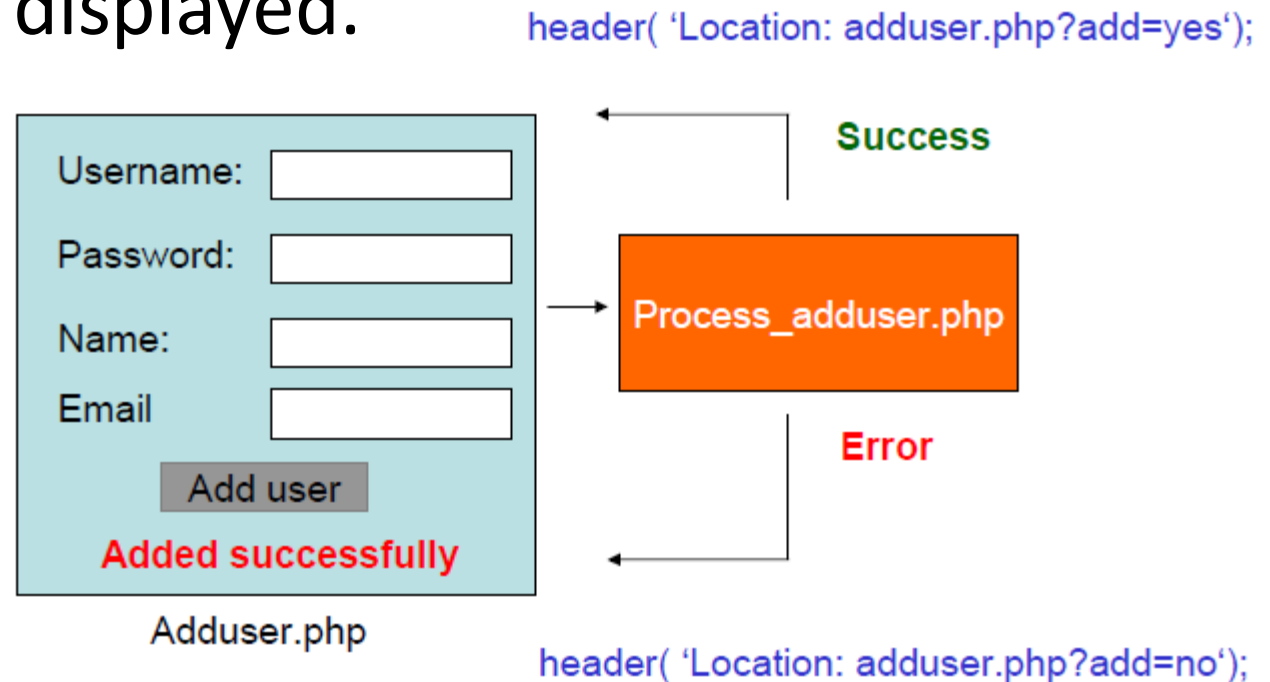
PHP Sessions

Aims of lecture

- At the end of today's lecture students should be able to
 - Connect to a mysql database using php
 - Retrieve, Insert, update and delete data from a mysql database using php
 - Use sessions

Redirecting to different pages

- A new user is being registered
- User is redirected to same page but different messages displayed.



Code to redirect and send info

```
//code to insert
```

```
//if insert successful –
```

```
header( 'Location: adduser.php?add=yes' );
```

```
else{
```

```
header( 'Location: adduser.php?add=no' );
```

```
}
```



Some useful functions

- Three useful functions to test for a variable are **isset()**, **empty()** and **is_null()**.
- All these function return a boolean value.
- **isset** — Determine if a variable is set and is not NULL, returns true only when the variable is not null and has been created.
- **empty** — Determine whether a variable is empty, it will return true if the variable is an empty string, false, array(), NULL, 0
- **is_null** — Finds whether a variable is NULL, it returns true only when the variable is null.

Capturing the value passed through the URL

- The value that has been passed through url is captured.

```
.... HTML codes to display
adduser controls
<?php
If ( isset($_GET['add']) ) {

    If ( $_GET ['add'] == "yes" )
        echo "Added successfully";
    Elself ( $_GET ['add' ] == "no" )
        echo "User already exists";
}
?>
....HTML codes
```



Username:

Password:

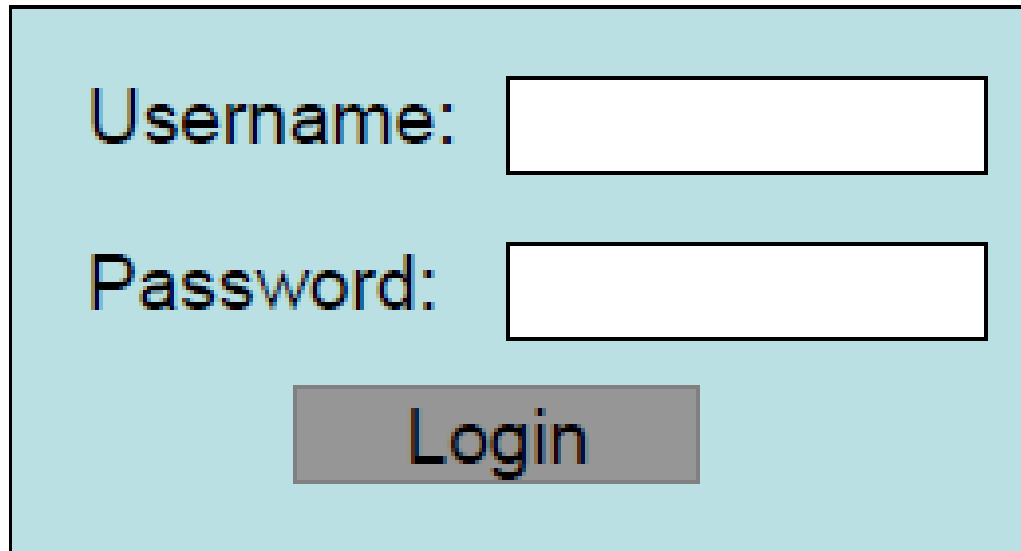
Name:

Email

Added successfully

Adduser.php

Example 1: login.html



A login form with a light blue background. It contains two labels, "Username:" and "Password:", each followed by a white input field with a black border. Below the input fields is a gray button with the text "Login".

Username:

Password:

Login

login.html

Login.html

- Consider the login.html page given above, when the page is submitted all information is submitted to checklogin.php.
- `<form name="frmlogin" method="post" action="checklogin.php">`
- The username and password captured on the login.html page is checked against saved values in the database and user is redirected to secondpage.php if all is ok.
- Write the code for login.html and checklogin.php

Checklogin.php(incomplete)

```
<?php
```

```
//missing code to capture username and password from previous page and save as $username and $pwd
```

```
//missing code to connect to mysql database and select database 'registered_users'
```

```
$sql = "Select * from login where username='$username' and password='$pwd'";
```

```
//missing code to run query and catch any possible errors
```

```
if (mysqli_num_rows($result) > 0) {  
header("Location:secondpage.php");
```

```
}
```


```
//missing code to close connection
```

```
?>
```

Session

- A session is a way to store information (in variables) to be used across multiple pages.
- Unlike a cookie, the information is not stored on the users computer.
- Session variables solve this problem by storing user information to be used across multiple pages (e.g. username, favorite color, etc). By default, session variables last until the user closes the browser.

Creating a session variable

- A session is started with the `session_start()` function.
- Session variables are set with the PHP global variable: `$_SESSION`.
- `$_SESSION['username'] =` 
`$_POST['txt_name'];`
- `//using the value that the person has typed in`

Adding session code to scenario

```
<?php
// Start the session
session_start();
....
//code to check username and password are ok 💬
//if both ok –
if (mysqli_num_rows($result) > 0) {   💬
$_SESSION['username'] = $_POST['txt_name'];
header( 'Location: secondpage.php' );
//we never reach here – we have already gone to details.php
else{
header( 'Location: login.html' );
//we never reach here – we have already gone to login.php
}
```

Check if user is logged in on all pages

```
<?php
// Start the session
session_start();
if(isset($_SESSION['username'])){
//show content of page to user
}
else{
header( 'Location: login.html' );
}
?>
```



Destroying a session

- `session_unset()` and `session_destroy()` are used to remove all global session variables and destroy the session

- ```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
```

```
<?php
// remove all session variables
session_unset();
```

```
// destroy the session
session_destroy();
?>
```

```
</body>
</html>
```

# Example – creating a session variable on 1 page

```
<?php
// Start the session
session_start();
?>
<!DOCTYPE html>
<html>
<body>
```

```
<?php
// Set session variables
$_SESSION["favcolor1"] = "black";
$_SESSION["favcolor2"] = "red";
echo "Session variables are set.";
?>
```

```
</body>
</html>
```

# Example – accessing session variable on another page

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
```

```
<?php
// Echo session variables that were set on previous page
echo "First Favorite color is " . $_SESSION["favcolor1"] . "
";
echo " Second Favorite color is " . $_SESSION["favcolor2"] . ".";
?>
```

```
</body>
</html>
```



# References

- My previous lecture notes
- [Http://www.w3schools.com](http://www.w3schools.com)
- <http://www.html5rocks.com>

Questions?