# EECS 485
(Web Databases & Information Systems)

Discussion

Jan 22th, 2010

## PA2

- Due on Monday
- Issues
  - Forum -> Phorum.eecs.umich.edu
  - AlbumAccess (albumid , username)
  - GD graphics library
- PA1 grades on Ctools; grade sheet in home dir.

## PA3

- Will be on Ctools on Monday, January 25
- Due on Wednesday, February 3.
- Authentication and Access Control on the album website from PA2.
- Don't modify PA2 (make a copy to PA3)

## PA3

- Personalizing the album website
- add a login page to the site
- use cookies to determine who the logged in user is
- Some pages are sensitive and require users login
- A user will have to explicitly log out, or allow the cookie to expire in order to no longer be authenticated

## PA3

- You'll implement actual sessions in a future project.
- You'll need simple Javascript for form validation.

## A bit about Javascript

PA3

## Javascript

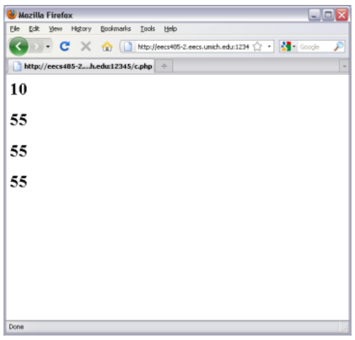
- Code between
  - `<script type="text/javascript">     </script>`
- Or external scripts
  - `<script type="text/javascript" src="xxx.js"> </script>`
- Statement `document.write("<h1>Hello Class</h1>");`
  - Semicolon is optional (multiple lines in 1 line)
- Comments are like PHP

## Variables

- Declare using var
  - `var x;`
    `var carname;`
- Case sensitive
- must begin with a letter or the underscore
  - `var x=5;`
    `var carname="BMW";`
- String concatenation with +
- exactly equal to (value and type)
  - `x===5 is true`
    `x==="5" is false`

## Example

```
<html>
<script type="text/javascript">
x=5+5;
document.write("<h1>"+x+"</h1>");
x="5"+"5";
document.write("<h1>"+x+"</h1>");
x=5+"5";
document.write("<h1>"+x+"</h1>");
x="5"+5;
document.write("<h1>"+x+"</h1>");
</script>
</html>
```


Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://eecs485-2.eecs.umich.edu:1234
Done

```
10
55
55
55
```

## Functions

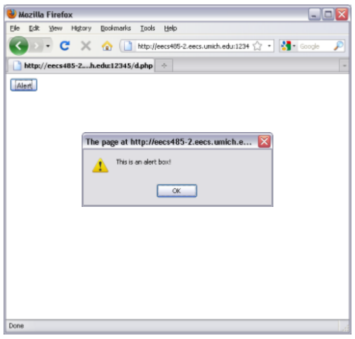
- To keep the browser from executing a script when the page loads, use functions.
- You may call a function from anywhere within a page (or even from other pages if the function is embedded in an external .js file).
- can be defined both in the <head> and in the <body> section of a document.
  - To assure that a function is read/loaded by the browser before it is called, better to define in <head>

## Alert Box

```
<html>
<head>
<script type="text/javascript">
function show_alert()
{
alert("This is an alert box!");
}
</script>
</head>
<body>

<input type="button" onclick="show_alert()"
value="Alert" />

</body>
</html>
```

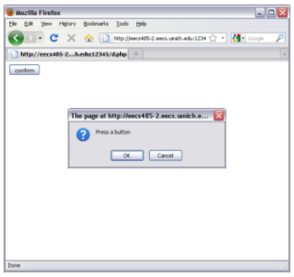

Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://eecs485-2.eecs.umich.edu:1234
Google
http://eecs485-2....h.edu:12345/d.php
Alert
OK
Done

The page at http://eecs485-2.eecs.umich.e...

This is an alert box!

## Confirm Box

```
<html>
<head>
<script type="text/javascript">
function show_confirm()
{
var r=confirm("Press a button");
if (r==true){
  document.write("You pressed OK!");
}else{
  document.write("You pressed Cancel!");
}
}
</script>
</head>
<body>
<input type="button" onclick="show_confirm()"
value="Alert" /> </body> </html>
```


Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://eecs485-2.eecs.umich.edu:1234
Google
http://eecs485-2....h.edu:12345/d.php
confirm
OK
Cancel
Done

The page at http://eecs485-2.eecs.umich.e...

Press a button

## Events

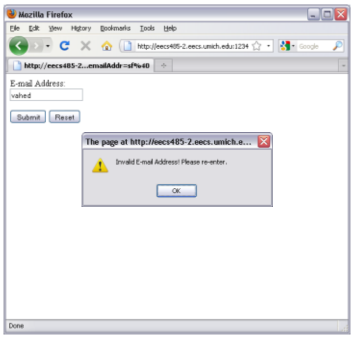
- Events are actions that can be detected by JavaScript.
- Every element on a web page has certain events which can trigger a JavaScript.
- Examples of events:
  - A mouse click
  - A web page or an image loading
  - Mouse over a spot on the web page
  - Selecting an input field in an HTML form
  - Submitting an HTML form
  - A keystroke

## Examples

- onChange often used for form validation
  - `<input type="text" size="30" id="email" onchange="checkEmail()">`
- Check form before submit
  - `<form method="post" action="xxx.htm" onsubmit="return checkForm()">`
- Other events
  - `onLoad and onUnload`
  - `onFocus, onBlur`
  - `onMouseOver and onMouseOut`

## Example

```
<HEAD>
<SCRIPT LANGUAGE="JavaScript">
function checkEmail(str) {
        if(str.indexOf("@") > 0){
           return true;
        }else{
           alert("Invalid E-mail Address!.")
           return (false);}
}
</script> </HEAD> <BODY>
<form onSubmit="return checkEmail(this.emailAddr.value)">
E-mail Address:<br> <input type="text" name="emailAddr">
<p> <input type="submit" value="Submit">
<input type="reset" value="Reset">
</form></body>
```


Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://eecs485-2.eecs.umich.edu:1234
Google
http://eecs485-2...emailAddr=sf%40
Submit
Reset
OK
Done
E-mail Address:

vahed

The page at http://eecs485-2.eecs.umich.e...

Invalid E-mail Address! Please re-enter.

## Session handling in PHP

PA3

## Sessions & Cookies

- HTTP communication is inherently stateless
- The way to handle state information is through sessions and cookies.
- PHP offers a built in mechanism for maintaining session information (e.g. hiding the cookie handling from the developer)

## Starting a Session

- Before you can begin storing user information in your PHP session, you must first start the session.
- it must be at the very beginning of your code, before any HTML or text is sent.
- Using
  - 
    ```
    <?
     session_start();
    ?>
    ```

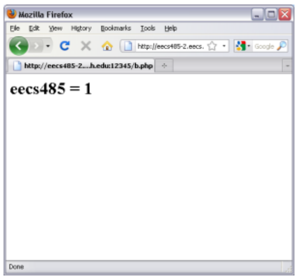
## Storing User Data

- The $_SESSION associative array is used to store user data.
- Used both to store and retrieve session data.
- $_SESSION[“views”] = 2
- $_SESSION[“name”] = “eecs”;

## Example

```
<?php
session_start();
$_SESSION['eecs485'] = 1; // store session data
?>

<html>
<h1>
<?
echo "eecs485 = ". $_SESSION['eecs485'];
//retrieve data
?>
</h1>
</html>
```



Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://eecs485-2.eecs.
http://eecs485-2....h.edu:12345/b.php
Done

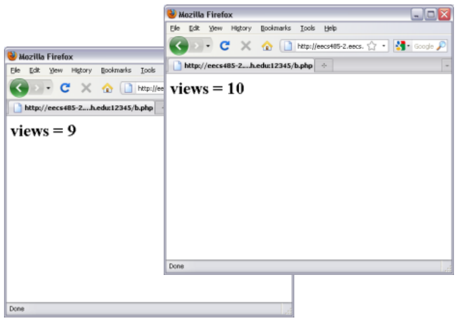
eecs485 = 1

## isset, unset

- Before you use a session variable it is necessary that you check to see if it exists already!
- `isset` is a function that takes any variable you want to use and checks to see if it has been **set**.
- You may wish to delete some data for your various tasks using `unset`

## Example

```
<?php
session_start();
if(isset($_SESSION['views']))
    $_SESSION['views'] = $_SESSION['views']+ 1;
else
    $_SESSION['views'] = 1;
?>
<html><h1>
<?
echo "views = ". $_SESSION['views'];
?>
</h1></html>
```



Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://eecs485-2....h.edu:12345/b.php
Done

views = 9

views = 10

## Destroying Sessions

- You can also completely destroy the session entirely by calling the `session_destroy` function.

```
<?
  session_destroy();
?>
```

- Destroy will reset your session, so don't call that function unless you are entirely comfortable losing all your stored session data!

## PHP Cookies

- create a cookie, using `setcookie`
- you must specify three arguments.
  - **name**: The name of your cookie. You will use this name to later retrieve your cookie, so don't forget it!
  - **value**: The value that is stored in your cookie.
  - **expiration**: The date when the cookie will expire and be deleted.
    - If you do not set this expiration date, then it will be treated as a session cookie and be removed when the browser is restarted.

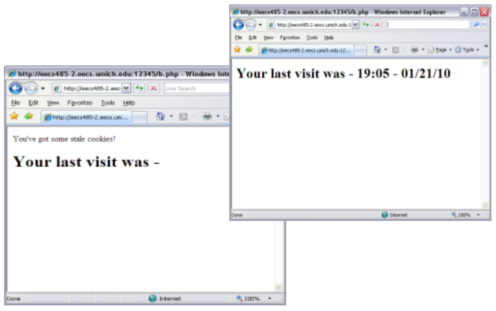
## Retrieve data

- If your cookie hasn't expired yet, you can retrieve it using `$_COOKIE` associative array.
- The name of your stored cookie is the key and will let you retrieve your stored cookie value!

## Example

```
<?php
$inOneDay = 60 * 60 * 24 + time();
setcookie('lastVisit', date("G:i - m/d/y"),
$inOneDay);
?>

<?php
if(isset($_COOKIE['lastVisit']))
    $visit = $_COOKIE['lastVisit'];
else
    echo "You've got some stale cookies!";
echo "Your last visit was - ". $visit;
?>
```


http://eecs485-2.eecs.umich.edu:12345/b.php - Windows Internet Explorer

Your last visit was - 19:05 - 01/21/10

You've got some stale cookies!

Your last visit was -