

# Information Technologies

**Anselm Spoerri** PhD (MIT)

SC&I @ Rutgers University

[aspoerri@rutgers.edu](mailto:aspoerri@rutgers.edu)

[anselm.spoerri@gmail.com](mailto:anselm.spoerri@gmail.com)

# Lecture 11 - Overview

## **PHP**

- Overview
- \$variables
- Strings
- Arrays
- Function
- IF test
- FOR loop

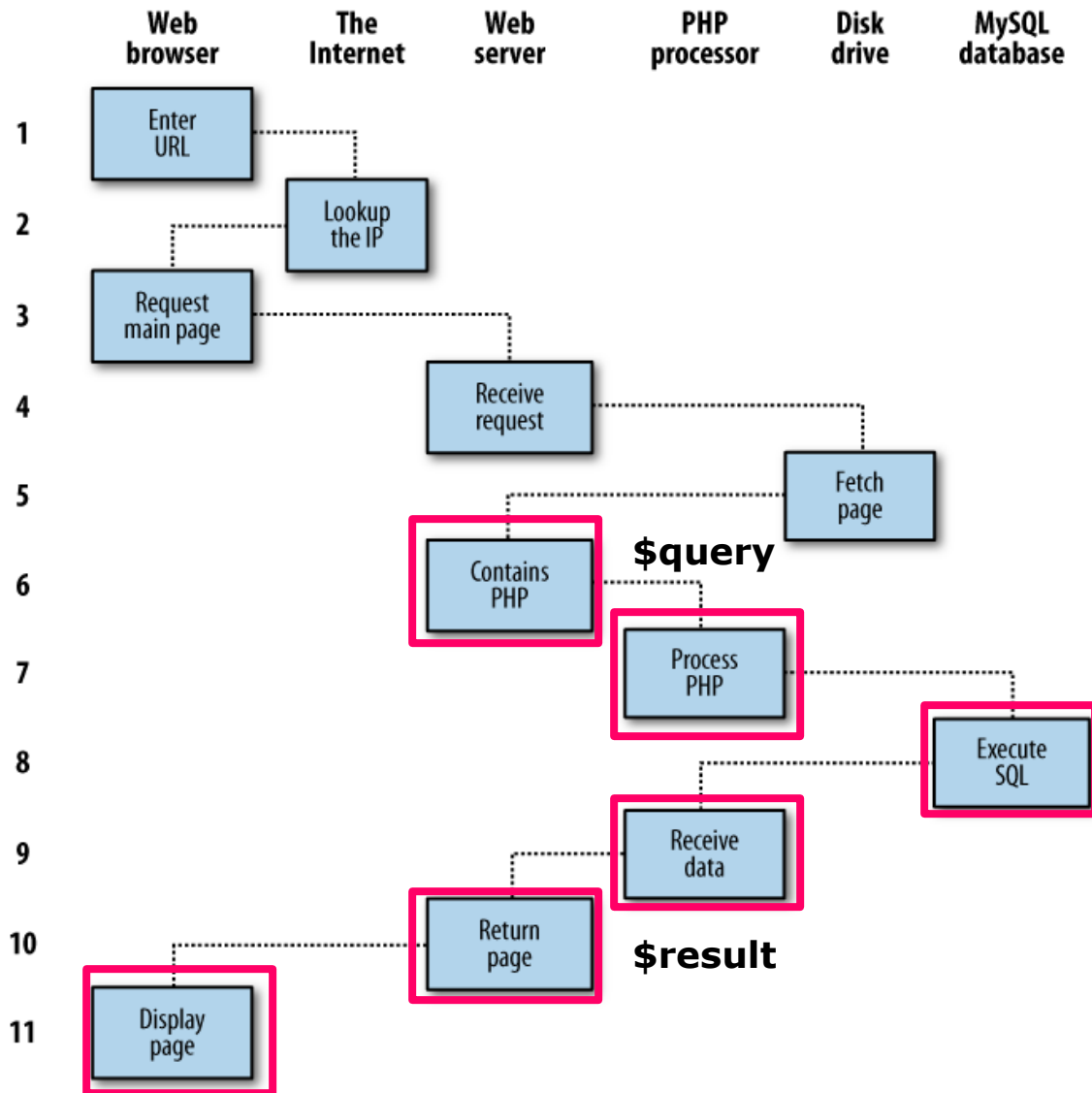
## **MySQL & PHP**

- login.php
- Connect to Your MySQL Database
- Run Query & Process Results

## **Lectures – Week 11 Content**

<http://comminfo.rutgers.edu/~aspoerri/Teaching/InfoTech/Lectures.html#week11>

# Dynamic Client/Server Request/Response Sequence



# Recap – Dynamic Web

## 1. JavaScript

- **Client-side scripting** access to elements of HTML document
- Perform data handling and web server requests in background

## 2. PHP

- **Server-side scripting**
- Open source, simplicity and built-in links to MySQL database

## 3. MySQL

- Open-source **Relational Databases** that supports **structured queries** and is free to use and install on web servers

## 4. Apache Server

- Open source server software that serves up HTML, media files etc

# PHP – Overview

<http://en.wikipedia.org/wiki/PHP>

**Open-source** <http://www.php.net> : documentation

## Server-side scripting

- **Insert content** of one file (or the output of a system call) into another file **dynamically**
- **Create images** and other files for you, either on the fly or in advance to serve up later.
- PHP program responsible for passing back clean **file suitable for display in web browser**.

## Simplicity & Flexibility

## Built-in links to MySQL database

**.php** file extension

```
<?php
    echo "Hello World. Today is ".date("l").".";
?>
```

## PHP – Overview (cont.)

```
<?php
```

```
    // This is a comment or /* multiline comments */
```

```
?>
```

```
$count = 10;
```

**Must Place \$ in front of all Variables**

**Common Error = Forget Semicolon**

```
$username = "Fred Smith"; // quotation marks for string
```

```
echo $username; // echo makes variable visible in Browser
```

## PHP – Code Construction

**HTML tags included inside of PHP code** (thus have DOCTYPE declaration and basic HTML structure)

Some programmers **open tag at start** of document and **close it at the end**, outputting any HTML directly from PHP commands.

```
<?php  
    echo "Hello World. Today is ".date("l").".";"  
?>
```

Others choose to **insert only smallest possible fragments of PHP** within these tags wherever dynamic scripting is required, leaving the rest of the document in standard HTML.

Hello World. Today is **<?php echo date("l"); ?>**.

Latter style may produce faster code, while the former has less complexity having to drop in and out of PHP many times in a single document.

# PHP - Variable Naming Rules

Starts with **\$**

## Variable Naming Rules

When creating PHP variables, you must follow these **four rules**:

- Must **start** with a **letter** of alphabet or **\_** (underscore)
- Can contain only the characters: **a-z, A-Z, 0-9, and \_**
- **May not contain spaces.**  
(e.g., \$user\_name).
- Variable names are **case-sensitive.**  
The variable \$High\_Score is not the same as the variable \$high\_score.



# PHP - Strings

PHP supports **two types of strings**: "hello" or 'hello'

- **Literal string** (preserve exact contents), **use apostrophe**:  
`$info = 'Preface variables with a $ like this: $variable';`
- **Include value of variable** (evaluate string): **use quotation marks**

```
$count = 44;
```

```
echo "There have been $count presidents of the US";
```

## Escape Character : \

- `$text = 'My friend\'s car is a Chevy';`
- `$text = "My mother always said \"Eat your greens\".";`
- **Important to use** when

**HTML code** and **variables** used inside **echo** statement

## PHP – Array Operations

```
$team = array('Bill', 'Joe', 'Mike', 'Chris'); // apostrophes  
echo $team[3]; // Displays the name Chris
```

→ **array indexes start at ZERO**

```
$twoDim = array(array('a', '', 'c'), array('', 'b', ''));  
echo $twoDim[1][1];
```

**count(\$a);** // number of array items      **Array union:** \$a + \$b

### **Associative Array**

– \$a['firstname'] = "John";

# PHP - Operators

## Operators

`++$j` or `--$j` // Increment or Decrement

`"FirstName" . " " . "LastName"` // String concatenation

**Boolean AND: && AND**

`$firstname && $lastname`

**OR: || OR**

`$firstname || $lastname`

**Equality Operator** is `==`    **Not Equal** is `!=`

**Identity Operator** is `===` (prevents type conversion)

Operators precedence rules → **use parentheses** to control what gets computed first and in which order

# PHP – Constants, Global and Function

## Define Constant

```
define("ROOT_LOCATION", "/usr/local/www/");
```

- **Can not** be prefaced with \$ sign and need to use define function
- Use UPPERCASE

## global \$username;

## Define Function

```
function function_name ([parameter [, ...]]) { // Statements }
```

```
function squared ($count) { return $count * $count;}
```

### Scope of Variable

- **Local variables** are accessible just from part of code where defined:

- Defined inside** of function, only **accessible inside** of function.

- Defined outside** of function, only **accessible outside** of function.

- **Global variables** are accessible from all parts of your code.

**Forgetting Scope of Variable is Common Error.**

## PHP – IF test

**if** (test) {do1;} **else** {do2;}

```
<?php
$a = "1000";
$b = "+1000";
if ($a == $b) {echo "equal";} else { echo "not equal";}
if ($a === $b) {echo "identical";} else { echo "not identical";}
?>
```

test ? doifTRUE : doifFALSE;

**if** (test1) {do1;} **elseif** (test2) {do2;} **else** {do3;}

# PHP – FOR Loop AND FOREACH Loop

## **for** loop

```
for (init variable ; test variable ; increment variable) {  
    do something;  
}
```

```
for ($count = 1 ; $count <= 12 ; ++$count) {  
    echo "$count times 12 is " . $count * 12 . "<br />";  
}
```

## **foreach** loop used for arrays

```
$paper = array("Copier", "Inkjet", "Laser", "Photo");  
$j = 0;  
foreach ($paper as $item) {  
    echo "$j: $item<br />";  
    ++$j;  
}
```

## PHP – Include Code from Other Files

### **include\_once "filename";**

- Tell PHP to fetch a particular file and load all its contents
- To avoid error of trying to define same constant or function multiple times

### **require\_once "filename";**

- To make sure that file is found

# MySQL & PHP – Process Steps

The process of using **MySQL** with **PHP**:

- 1. Connect to server** hosting your MySQL database(s)
- 2. Select database** to use.
- 3. Specify query** string.
- 4. Run query.**
- 5. Retrieve results** and **output it** to a web page.
- 6. Repeat Steps 3 to 5** until all desired data retrieved.
- 7. Disconnect from MySQL** (usually done automatically)



## MySQL & PHP – login.php for your database

Create login\_**lastname**.php file

```
<?php // login_lastname.php  make sure to place in personal folder
```

```
    $db_hostname = 'localhost';
```

```
    $db_database = 'yourFTPusername_mi550';
```

```
    $db_username = 'yourMySQLusername'; // yourFTPusername
```

```
    $db_password = 'yourMySQLpassword'; // yourFTPpassword
```

```
?>
```

Upload to Server, Set Permissions, View Source in Browser

Now that you have **login.php** file saved, you can include it in any PHP files that will need to access the database by using the **require\_once** statement.

# MySQL & PHP – Connect to Database – Step 1

```
<?php // connectDatabase.php
```

```
require_once 'login_lastname.php';
```

```
$db_connect = mysqli_connect($db_hostname,  
$db_username, $db_password);
```

```
if (!$db_connect) die("Unable to connect to MySQL: " .  
mysqli_error($db_connect));
```

```
mysqli_select_db($db_connect, $db_database) or die("Unable  
to select database: " . mysqli_error($db_connect));
```

```
?>
```

## MySQL & PHP – Query Database – Step 2

Add to connectDatabase.php

```
$query = "SELECT * FROM classics";
```

```
$result = mysqli_query($db_connect, $query);
```

```
if (!$result) die ("Database access failed: " .  
    mysqli_error($db_connect));
```

## MySQL & PHP – Display Query Result – Step 3

```
$rows = mysqli_num_rows($result);

for ($j = 0 ; $j < $rows ; ++$j){
    // fetch row

    $row = mysqli_fetch_row($result);

    // Need to consult MySQL table
    // to identify correct index value for field

    echo 'Author: ' . $row[0] . '<br>';
    echo 'Title: ' . $row[1] . '<br>';
    echo 'Year: ' . $row[3] . '<br>';
    echo 'ISBN: ' . $row[5] . '<br><br>';
}
```

# MySQL & PHP – Connect to Database – Steps 1-3

```
<?php // connectDatabase.php
require_once 'login_lastname.php';
$db_connect = mysqli_connect($db_hostname, $db_username, $db_password);
if (!$db_server) die("Unable to connect to MySQL: " . mysqli_error($db_connect));
mysqli_select_db($db_connect, $db_database) or die("Unable to select database: " .
    mysqli_error($db_connect));
$query = "SELECT * FROM classics";
$result = mysqli_query($db_connect, $query);
if (!$result) die ("Database access failed: " . mysqli_error($db_connect));
$rows = mysqli_num_rows($result);
for ($j = 0 ; $j < $rows ; ++$j){
    $row = mysqli_fetch_row($result);
    echo 'Author: ' . $row[0] . '<br>';
    echo 'Title: ' . $row[1] . '<br>';
    echo 'Year: ' . $row[3] . '<br>';
    echo 'ISBN: ' . $row[5] . '<br><br>';
?>
```

<http://comminfo.rutgers.edu/~aspoerri/Teaching/InfoTech/Lectures/Lec11/Steps/>

# Credential Management

## MySQL Workbench

Connection Method: **Standard TCP / IP**

Hostname: your**NetID**username.**rutgers-sci.domains**

Username: your**MySQL**username (= yourFTPusername or yourFTPusername\_user1)

Password: your**MySQL**password (= yourFTPpassword)

→ Can talk to **server with MySQL database** from **your computer**

## FTP

Connect to **rutgers-sci.domains** using your **FTP credentials**  
**yourFTPusername** and **yourFTPpassword**

→ Can **upload** HTML and PHP pages to **server** with **MySQL database**

→ Set **permissions**

## PHP

– Create login\_**your**lastname.php file that **stores MySQL credentials**  
so we can access MySQL from inside PHP code