Lesson Number 1

Name:

Introduction to Server Side Programming

Description:

Who am I?

- Programming since 2016
- Started with PHP
- Learned PHP by building a catalogue website for Sun City RV
- Built a few more sites, than gave it up to pursue a career in low voltage installations
- In 2010 had a nasty accident, so returned to school to reskill
- Built a few sites while studying. Started with WordPress, finished with PHP Laravel
- In 2013 graduated with honours in the Web Program at Georgian
- Learned Ruby and Rails in 2014 (favourite language and framework so far)
- Worked for GShift and freelanced for 2 years
- In late 2015, I left GShift to teach at Georgian
- Taught Introduction to Programming, then Bitmap & Vector

Introducing You

1. What is your name?
2. Where are you from?
3. Have you programmed with PHP before?

Class by Class Structure

00:00 - Previous Lesson Review

00:10 - Housekeeping

00:15 - Lesson

01:00 - Break

01:15 - Lesson Cont...

01:45 - Lesson Example

02:00 - Break

02:15 - Lesson Example Cont...

02:30 - Lesson Lab **(5%)**

Introduction to Class Tools

The Textbook, PHP/MySQL 2nd Edition

https://www.murach.com/shop/murach-s-php-and-mysql-2nd-edition-detail

PHP Documentation

http://php.net/manual/en/

MySQL Documentation

https://dev.mysql.com/doc/refman/5.7/en/

Choosing an IDE

https://www.jetbrains.com/student/
https://www.sublimetext.com/
https://netbeans.org/features/php/

Using Dreamhost

You Dreamhost account allows you to deploy your PHP applications. Deployment is achieved by either SFTP or by linking a GIT account and pulling your files to Dreamhost. The latter can be achieved by following the steps in this link:
http://wiki.dreamhost.com/Git

Using Filezilla & SFTP

Follow these steps to connect to Dreamhost using FileZilla:

1. Go to **File -> Site Manager**
2. Click **New Site**
3. Give your new site a name (like **Dreamhost**)
4. Under **Host**, enter **computerstudi.es**
5. Under **Port**, enter **22**
6. Under **Protocol**, choose **SFTP - SSH File Transfer Protocol**
7. Change **Logon Type**, to **Normal**
8. Enter the **User**name you were provided
9. Enter the **Password** you were provided
10. Once complete, click **Connect** to test your settings
11. Next time you open **FileZilla**, you can quick connect using the **downarrow** under the **Site Manager** icon, and selecting your site

Using MySQLWorkbench

Follow these steps to connect to your MySQL Database through Dreamhost

1. Go to **Database -> Manage Connections**
2. Click **New** to create a new connection
3. Title your new connection (like **Dreamhost**)
4. Change **Hostname** to **sql.computerstudi.es**
5. Leave **Port** as **3306**
6. Enter the **Username** you were provided
7. Click **Test Connection**
8. Enter the **Password** you were provided (choose **Save in Vault** to avoid having to type it

in everytime you connect)

Github Repository for Course

https://github.com/shaunmckinnon/georgian

Introducing Client/Server Architecture

Explanation

Servers

- they share resources such as files, printers, websites, databases, and email to clients
- servers can operate on every operating system
- some of the most common web servers are Apache VS Nginx
    - Apache
    - Nginx
    - IIS
- a web server shares websites to a client's web browser

Clients

- clients allow users to interact with server content
- clients recieve and output/mutate the information from the server
- common clients are
    - email applications
    - web browsers
    - online/multiplayer games
    - cloud based applications
    - database applications
    - home media interfaces

Networks

- a network is a communication system that allows servers and clients to communicate with one another
- routing is a process in which information is transferred from one computer to another
- a router is a device that connects to two or more networks and handles the flow of information

Local Area Network

- LAN stands for Local Area Network
- a LAN is a small network of computers usually within the same building
- LAN is also sometimes known as an intranet
- LANs will sometimes host local web applications that are accessible by LAN connected users only

Wide Area Networks

- WAN stands for Wide Area Network
- a WAN consists of multiple LANs connected together through routers

Internet Service Provider

- ISP stands for Internet Service Provider
- An ISP is a company that owns a WAN that is connected to the internet
- An ISP leases access to its network giving user access to the internet

The Internet

- the internet is a global network consisting of multiple WANs that have been connected together
- Internet Exchange Points (IXP) connect ISP's WANs together providing the ability to exchange information with anyone connected IT Crowd (Jen and the Internet) 3:00

Anatomy of a Static Web Page

How Static Web Pages are Processed

1. User requests a web page in their web browser (either by typing a link in the address bar or by clicking a link)
2. A request is built by the web browser and sent to the web server
   NOTE: this request is known as an HTTP Request
3. The HTTP Request contains the following information Anatomy of an HTTP Request
   1. The request type (GET, POST, PUT, DELETE, HEAD, TRACE, CONNECT) Request Types
   2. The file being requested (/index.html, directory/index.php)
   3. The HTTP version being used (HTTP/1.1)
   4. The host address that has the file we're requesting (www.example.com)
4. When the hosting server receives the request, it checks the requested file's extension to establish which program or server should process the request
5. Once the request has been processed it the host server returns the requested file as an HTTP Response
6. The HTTP Response contains the following information:
   1. The HTTP Response Status (HTTP/1.1 200 OK) HTTP Response Codes
   2. The Content Type (MIME type - Multipurpose Internet Mail Extensions) (text/html, application/x-doom, image/png) MIME Types
   3. Content-Length
   4. Server (Apache, Nginx, IIS)
   5. The actual content
7. This request and response process relies on two protocols
   1. HTTP - HyperText Transfer Protocol
      a protocol that web browsers and servers use to communicate. It sets the specificatons for HTTP requests and responses
   2. TCP/IP - Transmission Control Protocol/Internet Protocol
      a suite of protocols that let two computers communicate over a network

Anatomy of a Dynamic Web Page

How Dynamic Web Pages are Processed

1. User requests a web page in their web browser (either by typing a link in the address bar or by clicking a link)
2. A request is built by the web browser and sent to the web server
3. When the hosting server receives the request, it checks the requested file's extension to establish which program or server should process the request
4. A dynamic web page will use script in order to generate a web page with data from a script, database, API, file, or another source
5. Once the request has been processed it the host server returns the requested file as an HTTP Response

Basic Dynamic Web Site Example

Example Objective

- explore the benefits of dynamic pages vs static pages
- learn PHP basic syntax
    - wrapping tags
    - comments
    - variables
    - output
    - requiring a file
- creating the default HTML starter page for all labs

Why

- static pages require thorough editing and you may miss key elements
- a static page has to be created for every scenario even if it is redundant changes
- a static page can only use clientside scripting to retrieve and format data, which can be circumvented by the user
- static pages are faster than dynamic pages but have less versatility
- dynamic pages provide access to a wide array of data sources (files, APIs, databases, cloud storage, scrapings, etc...)
- dynamic pages allow for autogenerated web pages for varying scenarios
- dynamic pages allow the capture of data

Quick Reference

```
// opening & closing PHP tags wrap your PHP code
```

```
<?php ?>

// single line comment

/* first line of the multiline comment
   second line of the multiline comment */

// outputting strings to the screen
echo "This is my String";

/* This method is a quick inline version
   of outputting to the screen. It's handy
   when adding dynamic elements to HTML. */
<?= "This is my String"; ?>

/* This is a variable. Variables can store
   varying data type including arrays */
$my_string = "Shaun";
$my_number = 37;
$my_float = 37.5;
$my_array = [6, 7, 8, 9];

/* All PHP statements have to be ended with a
   semicolon (;) as this tells the parser where
   the command ends. The only exception is comments. */

/* Including or Requiring files is common practice
   used to modularize code. */
require '/path/to/my/php/code.php'; // will throw a fatal
error if not found
include '/path/to/my/php/code.php'; // will throw a
warning, but continue to process the page
```

```
<!DOCTYPE HTML>
<html lang="en">

  <head>
    <title>title</title>
  </head>

  <body>
  </body>

</html>
```