A290/A590
CGI/PHP
Lecture Notes

including a study guide for the final

Dan-Adrian German with
Janani Sivakumar and
Preetesh Patodi

Last updated: Wednesday 31st January, 2018 at 04:39
http://silo.cs.indiana.edu:8346/a290-web/study-guide

Contents

Foreword

We believe that teaching is an art as well a science and a privilege of the utmost distinction. As teachers we hold the following truths to be self-evident:

(a) that all students are inherently motivated to learn, but they quickly learn to be unmotivated if they fail repeatedly;

(b) that every student has the basic need to belong, to be competent and to influence what happens to her/him; motivation to learn only exists when these three conditions are met;

(c) that learning is difficult, expensive and risky and therefore students must perceive the classroom as a safe environment, both physically and psychologically; in this respect high self-esteem should not be a goal, but a consequence of mastering the material.

The text you're reading is not to be printed since it is likely to be updated very often and can be easily accessed online[1]. We acknowledge the support and careful feedback received on this document from the following individuals to whom this material is in fact dedicated:

> Yuyao Ba, Seungmin Baek, Levi Beyers, Elissa Booras, Xinran Dai, Kwong Yuet Michael Fadilah Wong, Jacob Good, Sungwon In, Younghun Kim, Chad Kowaleski, Brandon Lengacher, Canrong Lin, Keyi Lin, Fereico Ortiz, Brandon Partin, Nicolas Petrovich Mora, Alex Queen, Pranav Rajendiran, Mitchell Walls, Matthew Wyrick, Yifan Zhang, and Yonjae Lee.

This set of lecture notes is meant to be used alongside the textbook(s) we recommended for the class. Please let us know[2] if anything is unclear or incorrect or if something seems to be missing.

[1] `https://www.cs.indiana.edu/~dgerman/a290-web/study-guide.pdf`
[2] `dgermanindiana.edu`


The Polar Express
BELIEVE
N° 0001225

Chapter 1

Unix

1.1 silo

Your host server is `silo.soic.indiana.edu` and it runs Linux[1].

```
-bash-4.2$ hostname
silo.soic.indiana.edu
-bash-4.2$ uname -o
GNU/Linux
-bash-4.2$ uname -snr
Linux silo.soic.indiana.edu 3.10.0-693.11.6.el7.x86_64
-bash-4.2$ uname -i
x86_64
-bash-4.2$
```

1.2 PuTTY

If you're running Windows you need to download PuTTY[2] and use it to connect to `silo`. Once you connect you are being presented with a prompt and you can start typing Unix commands.

```
login as: dgerman
dgerman@silo.cs.indiana.edu's password:
Last login: Tue Jan 30 22:10:05 2018 from [...]

-bash-4.2$ pwd
/u/dgerman
-bash-4.2$
```

[1] `https://en.wikipedia.org/wiki/Linux`
[2] `https://iuware.iu.edu/Windows/List/140`

1.3 Xming

Xming is free and supported by IU[3].

1.4 XQuartz

XQuartz[4] is also free and used for connections from your Mac.

1.5 clear

This command clears the screen.

1.6 cd

This command helps you change the current directory.

1.7 pwd

This command shows you the folder you're in.

1.8 mkdir

This helps create a folder.

1.9 ls

This shows the content of a folder.

1.10 nano

This starts the editor so you can author or edit a file.

1.11 touch

This updates the time stamp on a file. If the file does not exist an empty file with that name is created.

1.12 tree

This shows the structure of a folder.

[3] https://kb.iu.edu/d/bdnt
[4] https://www.xquartz.org/

1.13 `du`

This does almost the same thing as `tree`.

1.14 `ps`

This shows what processes are running.

1.15 `cp`

This copies a file or a folder.

1.16 `mv`

This moves (renames) a file.

1.17 `tar`

This creates or extracts an archive.

1.18 `gunzip`

This uncompresses a compressed file.

1.19 `rm`

This helps remove a file or folder.

1.20 `cat`

This shows the content of one or more files, concatenated.

1.21 `grep`

This looks for regular patterns in a text file.

1.22 `|`

This is the Unix pipe(line) character[5].

[5] `https://en.wikipedia.org/wiki/Pipeline_(Unix)`

1.23 `man`

This calls the Unix manual pages.

1.24 `chmod`

This changes the mode on a file or folder.

1.25 `umask`

This command helps set the user file-creation mask.

1.26 `whoami`

This reports the username who owns the session where the command is issued.

1.27 `hostname`

This reports the name of the machine you're connected to.

1.28 `who`

This shows the users currently logged on.

1.29 `talk`

This allows two users to communicate with each other in real time.

1.30 `telnet`

Unencrypted communication with another machine.

1.31 `quota`

Shows how you're doing spacewise on the system.

1.32 `makenobackup`

This is a command specific to `silo` and gives you more space on a partition that is not backed up.

1.33 `kill`

Helps stop a process.

1.34 `history`

Gives you a list of all commands you typed in the recent past.

1.35 `netstat`

Shows what's happening on the network interface.

1.36 `cron`

Helps set or schedule jobs.

1.37 Lab One

You need to create a hierarchy of files and folders, then pack it up in an archive. Put the archive in a location that is publicly available without jeopardizing the security of your account. The instructor will access and grade it remotely.

Chapter 2

Apache

Apache is a foundation. Apache HTTP is a web server.

2.1 Installation

Follow the steps to install Apache.

2.2 Password protected folders

Make sure you post your assignments (source code especially) here.

2.3 Starting/stopping

You need to know how to start/stop your Apache server.

2.4 Automatic restart

Set it up so the server restarts four times a day.

2.5 Lab Two

We need a server up with a page that states your name and shows a picture of something you are fond of.

Chapter 3

HTTP

This is a protocol, used between browsers and server.

3.1 GET

In this case the headers are followed by just an empty line.

3.2 POST

One of the headers states the size of the data that is sent after the empty line that comes at the end of the headers.

Chapter 4

HTML

Chapter 5

Perl

Chapter 6

Python

6.1 Readings

Chapter 7

CGI

CGI is a convention.

Chapter 8

State

8.1 Constants

8.2 Inputs

8.3 Outputs

8.4 State

Chapter 9

MySQL

9.1 Installation

9.2 Lab 03

Chapter 10

PHP

10.1 Installation

Install PHP after you install MySQL.

10.1.1 Lab 04

Turn in Lab 04 and Lab 03 at the same time.

10.2 PHP Crash Course

1. When[1] was PHP invented?
2. Who[2] was the inventor?
3. Create a form that looks like this[3]:

4. Write the PHP script[4] that handles input from it.
5. Describe the four distinct types[5] of PHP tags.
6. Write a PHP script[6] that prints the current date on the server.
7. What does it mean to filter your data? And why[7] should you do it?
8. What (basic) data types does PHP support[8]?

[1] See page 2 of your text.
[2] See page 2 of the text.
[3] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/001/one.html`
[4] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/001/two.phps`
[5] See pages 18-19 of the text.
[6] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/001/three.phps`
[7] Page 25.
[8] See page 29 in the text.

9. What is a variable[9] variable?

10. What are superglobals? Give[10] the complete list.

11. What is the effect of this statement[11]:

```
$b = 6 + ($a = 5);
```

12. Describe[12] how the reference operator works.

13. Name some array operators[13].

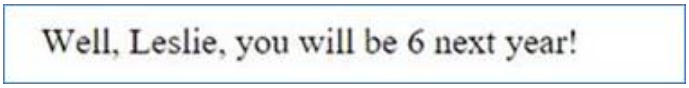
14. What are variable[14] functions?

15. Write a PHP script[15] to generate a table with the first ten Fibonacci[16] numbers. Solution[17].

And now a capstone question for this section: write a PHP script that combines questions 3 & 4 above. In other words, when you call it, it responds with:

Then you type a name and an age in it:

Now if you push the submit button you get:



But if you don't provide a name or an age or if you call it again:

Here's how[18] it should behave, essentially. Try it out.

---

[9] See page 30.
[10] See page 32 in text.
[11] See page 34.
[12] See page 36.
[13] See page 40.
[14] See pages 44-46.
[15] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/001/four.php`
[16] `https://en.wikipedia.org/wiki/Fibonacci_number`
[17] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/001/four.phps`
[18] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/001/question.php`

10.3 Storing and Retrieving Data

1. What do we mean by a flat file?
2. What[19] other ways of storing and retrieving data exist?
3. Write a script[20] that adds[21] the current time to a file every time it's run.
4. Write a script[22] that counts the number of lines in a file and reports it.

And now, the question:

> Use the code developed above to write a personal homepage (sic!) that keeps track of how many times it's been accessed.

Here's how[23] it should work.

10.4 Using Arrays

1. Create a numerically indexed array in PHP.
2. Create an associative array in PHP.
3. Create a multidimensional array in PHP.
4. Use a pre-existing function to print the array.
5. Use a loop to print the array.
6. How can you load an array from a file?

And now, the question:

> Give a meaningful example of using `extract()`.

See the online[24] documentation.

---

[19] Page 59.
[20] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/002/one.phps`
[21] Pages 62-69.
[22] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/002/two.phps`
[23] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/002/three.php`
[24] `http://php.net/manual/en/function.extract.php`

10.5 String Manipulation and Regular Expressions

1. Give a few examples of how and when we need to format strings in PHP.
2. How do you join and/or split strings in PHP?
3. How do you match and replace substrings in PHP?
4. What is a regular expression?
5. Why do we need to format strings for database storage?
6. Why do we need to escape strings? How/why/when do we filter inputs?

And now, the question:

> Give an example of using the `nl2br()` function.

Hint: search for "filter" in Chapter 11 (from pp. 268-9 on).

10.6 Reusing Code and Writing Functions

1. Create a simple view[25] whose purpose is to show variables `i` and `j`.
2. Create another simple view[26] that aims to show the values of `i` and `k`.
3. Create a third view[27] that displays values of variables `i`, `j` and `k`.
4. Create a controller[28] that assigns values to i, j, k then calls a random view to show them.

And now, the question:

> Apply the info above and develop a solution to this problem[29].

Here's how we could get started:

1. the controller[30] (`heart.php`)
2. one[31] of the views (`welcome.php`)
3. another[32] one (`great.php`)
4. and a third[33] one (`darn.php`)

Here's how the final programs looks and behaves[34].

[25] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/005/view01.phps`
[26] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/005/view02.phps`
[27] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/005/view03.phps`
[28] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/005/controller.phps`
[29] `https://www.cs.indiana.edu/classes/a290-web/sum2008/a290/example.html`
[30] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/005/exercise/heart.phps`
[31] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/005/exercise/welcome.phps`
[32] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/005/exercise/great.phps`
[33] `http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/005/exercise/darn.phps`
[34] `ttps://www.cs.indiana.edu/classes/a290-web/sum2008/a290/example.html`

10.7 Object-Oriented PHP

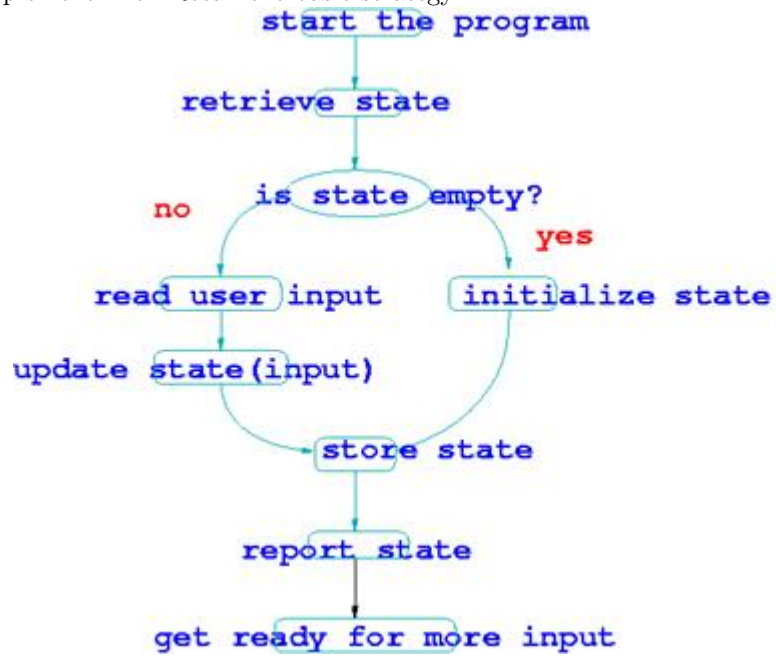
1. What is inheritance?
2. What is polymorphism?
3. What is a constructor?
4. What is overriding?
5. What is an interface?
6. What is an abstract class?

And now, the question:

> Where in Bob's Auto Parts are these concepts used?

Also: explain how these scripts[35] [36] [37] work[38].

Next, consider the flag[39] quiz and the following comparison[40]. Can you implement[41] it? Recall the basic strategy:



[35] http://silo.cs.indiana.edu:8346/a348/fall2014/template.phps
[36] http://silo.cs.indiana.edu:8346/0301/one.phps
[37] http://silo.cs.indiana.edu:8346/0301/two.phps
[38] They work together.
[39] https://www.cs.indiana.edu/classes/a348/fall2008/whatsnew/1113.html
[40] https://www.cs.indiana.edu/classes/a348/fall2008/whatsnew/1116.html
[41] http://silo.cs.indiana.edu:8346/a202/fall2017/hw09/stage000.html

Finally, describe the pertinence of this picture in the current context:

Figure 10.1: Please explain the relative significance of this picture in the current context. What does it show, if anything?

Before this chapter is over please consider the following:

1. here's a template[42]
2. here's an implementation[43] of it for a specific problem
3. here's the driver[44] for the entire program
4. we could move the entire state in a database[45]
5. and define a new driver[46]
6. finally consider this[47] and its conversion[48]

Show me these programs working on your server(s) and be ready to discuss and explain them.

[42] `http://silo.cs.indiana.edu:8346/template.phps`
[43] `http://silo.cs.indiana.edu:8346/client.phps`
[44] `http://silo.cs.indiana.edu:8346/one.phps`
[45] `http://silo.cs.indiana.edu:8346/server.phps`
[46] `http://silo.cs.indiana.edu:8346/secondTwo.phps`
[47] `http://silo.cs.indiana.edu:8346/main0420.phps`
[48] `https://www.cs.indiana.edu/classes/a202-dger/spr2010/0422.html`

10.8 Error and Exception Handling

1. What is an exception? (Or is it Exception?)
2. What is a user-defined exception?

And now, the question:

> Identify exceptions thrown/caught in our Bob's Auto Parts code.

10.9 Advanced PHP

10.9.1 Managing the Date and Time

10.9.2 Using Session Control in PHP

Chapter 11

Using MySQL

Chapter 12

E-commerce and Security

Chapter 13

Practical Projects

Chapter 14

The MEAN Stack

Chapter 15

Commencement

15.1 Quote of the Day