

A290/A590 CGI/PHP Lecture Notes

including a study guide for the final

DAN-ADRIAN GERMAN with
JANANI SIVAKUMAR and
PREETESH PATODI



© COPYRIGHT BY DAN-ADRIAN GERMAN.
© Copyleft too. ® All rights reserved worldwide.
Last updated: Wednesday 31st January, 2018 at 04:39
<http://silo.cs.indiana.edu:8346/a290-web/study-guide>

Contents

1	Unix	3
1.1	silob	3
1.2	PuTTY	3
1.3	Xming	4
1.4	XQuartz	4
1.5	clear	4
1.6	cd	4
1.7	pwd	4
1.8	mkdir	4
1.9	ls	4
1.10	nano	4
1.11	touch	4
1.12	tree	4
1.13	du	5
1.14	ps	5
1.15	cp	5
1.16	mv	5
1.17	tar	5
1.18	gunzip	5
1.19	rm	5
1.20	cat	5
1.21	grep	5
1.22	5
1.23	man	6
1.24	chmod	6
1.25	umask	6
1.26	whoami	6
1.27	hostname	6
1.28	who	6
1.29	talk	6
1.30	telnet	6
1.31	quota	6
1.32	makenobackup	6
1.33	kill	7

1.34	history	7
1.35	netstat	7
1.36	cron	7
1.37	Lab One	7
2	Apache	9
2.1	Installation	9
2.2	Password protected folders	9
2.3	Starting/stopping	9
2.4	Automatic restart	9
2.5	Lab Two	9
3	HTTP	11
3.1	GET	11
3.2	POST	11
4	HTML	13
4.1	html	13
4.2	head	13
4.3	body	13
4.4	p	13
4.5	a	13
4.6	img	13
4.7	form	13
4.8	input	13
5	Perl	15
5.1	Literals	15
5.2	Operators	15
5.3	Values	15
5.4	Types	15
5.5	Scalars	15
5.6	Assignments	15
5.7	Decisions	15
5.8	Loops	15
5.9	Lists	15
5.10	Arrays	15
5.11	Command-line Arguments	15
6	Python	17
6.1	Readings	17
7	CGI	19
8	State	21
8.1	Constants	21
8.2	Inputs	21

8.3	Outputs	21
8.4	State	21
9	MySQL	23
9.1	Installation	23
9.2	Lab 03	23
10	PHP	25
10.1	Installation	25
10.1.1	Lab 04	25
10.2	PHP Crash Course	25
10.3	Storing and Retrieving Data	27
10.4	Using Arrays	27
10.5	String Manipulation and Regular Expressions	28
10.6	Reusing Code and Writing Functions	28
10.7	Object-Oriented PHP	29
10.8	Error and Exception Handling	31
10.9	Advanced PHP	31
10.9.1	Managing the Date and Time	31
10.9.2	Using Session Control in PHP	31
11	Using MySQL	33
11.1	Database Design	33
11.2	Database Creation	33
11.3	Working with Your Database	33
11.4	Accessing Your Database from PHP	33
11.5	Accessing Your Database from CGI/Python	33
11.6	Accessing Your Database from CGI/Perl	33
11.7	Advanced MySQL Administration	33
11.8	Advanced MySQL Programming	33
12	E-commerce and Security	35
12.1	Running an E-commerce Site	35
12.2	E-commerce Security Issues	35
12.3	Web Application Security	35
12.4	Implementing Authentication with PHP and MySQL	35
12.5	Implementing Secure Transactions with PHP and MySQL	35
13	Practical Projects	38
13.1	Shopping Cart	38
13.2	Robin's Nest	38
13.2.1	Javascript	38
13.2.1.1	Validation from Javascript and PHP	38
13.2.2	Ajax	38
13.2.3	CSS	38
13.2.3.1	Accessing CSS from Javascript	38

13.2.4	HTML5	38
13.2.4.1	Canvas	38
13.2.4.2	Geolocation	38
13.2.4.3	Audio and Video	38
13.2.4.4	Forms	38
13.2.4.5	Local Storage	38
13.2.4.6	Web Workers	38
13.2.4.7	Web Applications	38
13.3	Django	38
13.4	CakePHP	38
13.4.1	Web Blog	38
13.4.2	QuickWall	38
14	The MEAN Stack	39
14.1	Node	39
14.2	Express	39
14.3	Mongo	39
14.4	Angular	39
15	Commencement	41
15.1	Quote of the Day	41

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 online¹. 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 know² 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>

²dgermanindiana.edu



Chapter 1

Unix

1.1 silo

Your host server is `silo.soic.indiana.edu` and it runs Linux¹.

```
-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 PuTTY² 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 IU³.

1.4 XQuartz

XQuartz⁴ 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) character⁵.

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

4.1 html

4.2 head

4.3 body

4.4 p

4.5 a

4.6 img

4.7 form

4.8 input

Chapter 5

Perl

5.1 Literals

5.2 Operators

5.3 Values

5.4 Types

5.5 Scalars

5.6 Assignments

5.7 Decisions

5.8 Loops

5.9 Lists

5.10 Arrays

5.11 Command-line Arguments

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

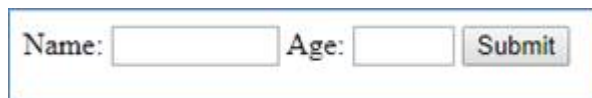
Install PHP after you install MySQL.

10.1.1 Lab 04

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

10.2 PHP Crash Course

1. When¹ was PHP invented?
2. Who² was the inventor?
3. Create a form that looks like this³:



The image shows a simple web form with a light blue border. It contains two text input fields. The first is labeled "Name:" and the second is labeled "Age:". To the right of the "Age:" field is a button labeled "Submit".

4. Write the PHP script⁴ that handles input from it.
5. Describe the four distinct types⁵ of PHP tags.
6. Write a PHP script⁶ that prints the current date on the server.
7. What does it mean to filter your data? And why⁷ should you do it?
8. What (basic) data types does PHP support⁸?

¹See page 2 of your text.

²See 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>

⁵See pages 18-19 of the text.

⁶<http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/001/three.phps>

⁷Page 25.

⁸See page 29 in the text.

9. What is a variable⁹ variable?
10. What are superglobals? Give¹⁰ the complete list.
11. What is the effect of this statement¹¹:


```
$b = 6 + ($a = 5);
```
12. Describe¹² how the reference operator works.
13. Name some array operators¹³.
14. What are variable¹⁴ functions?
15. Write a PHP script¹⁵ to generate a table with the first ten Fibonacci¹⁶ numbers. Solution¹⁷.

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:

Name:

Age:

Then you type a name and an age in it:

Name:

Age:

Now if you push the submit button you get:

Well, Leslie, you will be 6 next year!

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

Name:

Age:

Here's how¹⁸ it should behave, essentially. Try it out.

⁹See page 30.

¹⁰See page 32 in text.

¹¹See page 34.

¹²See page 36.

¹³See page 40.

¹⁴See pages 44-46.

¹⁵<http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/001/four.php>

¹⁶https://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. What¹⁹ other ways of storing and retrieving data exist?
3. Write a script²⁰ that adds²¹ the current time to a file every time it's run.
4. Write a script²² 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 how²³ 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 online²⁴ documentation.

¹⁹Page 59.

²⁰<http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/002/one.php>

²¹Pages 62-69.

²²<http://silo.cs.indiana.edu:8346/a290-web/spr2016/hw07/002/two.php>

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 `n12br()` function.

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

10.6 Reusing Code and Writing Functions

1. Create a simple view²⁵ whose purpose is to show variables `i` and `j`.
2. Create another simple view²⁶ that aims to show the values of `i` and `k`.
3. Create a third view²⁷ that displays values of variables `i`, `j` and `k`.
4. Create a controller²⁸ 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 problem²⁹.

Here’s how we could get started:

1. the controller³⁰ (`heart.php`)
2. one³¹ of the views (`welcome.php`)
3. another³² one (`great.php`)
4. and a third³³ one (`darn.php`)

Here’s how the final programs looks and behaves³⁴.

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<https://www.cs.indiana.edu/classes/a290-web/sum2008/a290/example.html>

10.7 Object-Oriented PHP

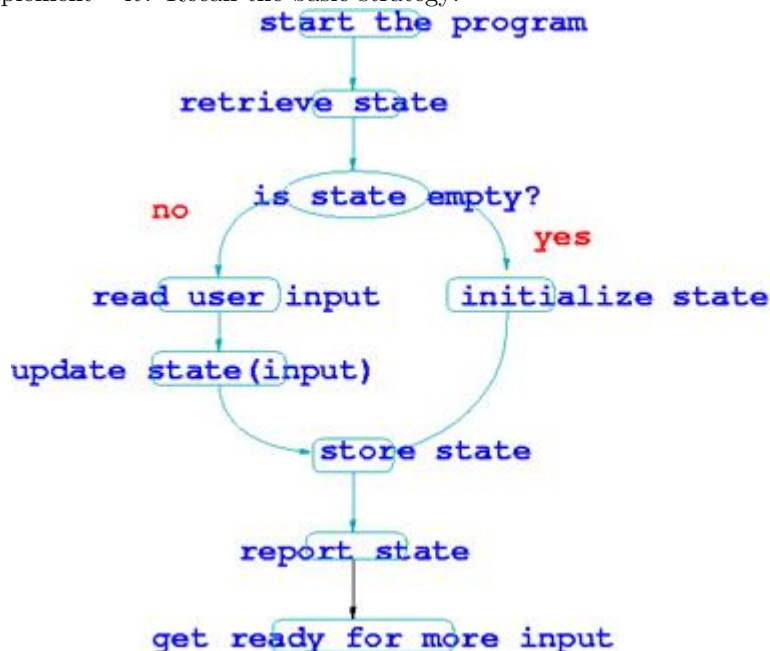
1. What is inheritance?
2. What is polymorphism?
3. What is a constructor?
4. What is overriding?
5. What is an interface?
6. What is an abstract class?

And now, the question:

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

Also: explain how these scripts^{35 36 37} work³⁸.

Next, consider the flag³⁹ quiz and the following comparison⁴⁰. Can you implement⁴¹ 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>

³⁸They 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 template⁴²
2. here's an implementation⁴³ of it for a specific problem
3. here's the driver⁴⁴ for the entire program
4. we could move the entire state in a database⁴⁵
5. and define a new driver⁴⁶
6. finally consider this⁴⁷ and its conversion⁴⁸

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

- 11.1 Database Design
- 11.2 Database Creation
- 11.3 Working with Your Database
- 11.4 Accessing Your Database from PHP
- 11.5 Accessing Your Database from CGI/Python
- 11.6 Accessing Your Database from CGI/Perl
- 11.7 Advanced MySQL Administration
- 11.8 Advanced MySQL Programming

Chapter 12

E-commerce and Security

- 12.1 Running an E-commerce Site
- 12.2 E-commerce Security Issues
- 12.3 Web Application Security
- 12.4 Implementing Authentication with PHP and MySQL
- 12.5 Implementing Secure Transactions with PHP and MySQL

Chapter 13

Practical Projects

13.1 Shopping Cart

13.2 Robin's Nest

13.2.1 Javascript

13.2.1.1 Validation from Javascript and PHP

13.2.2 Ajax

13.2.3 CSS

13.2.3.1 Accessing CSS from Javascript

13.2.4 HTML5

13.2.4.1 Canvas

13.2.4.2 Geolocation

13.2.4.3 Audio and Video

13.2.4.4 Forms

13.2.4.5 Local Storage

13.2.4.6 Web Workers

13.2.4.7 Web Applications

13.3 Django

13.4 CakePHP

13.4.1 Web Blog

13.4.2 QuickWall

Chapter 14

The MEAN Stack

14.1 Node

14.2 Express

14.3 Mongo

14.4 Angular

Chapter 15

Commencement

15.1 Quote of the Day