



Appendix A

Answers to Self Test

Chapter 1: Setting Up Your Workstation

1. Integrated development environment
2. No
3. Any five of the following: code assistance, code completion, code validation, browser view, outline view, line numbers, auto-indentation, search and replace, macro capability, multiple open files, multiple language support, file management, FTP transfer, and version control
4. Code assistance suggests code as you type; code validation points out potential errors in your code.
5. Primarily Aptana Studio and Zend Studio, but could also be Komodo IDE or PHPStorm
6. A version-control and source-code app
7. Client-side scripts run solely in a browser; server-side scripts run in the server instead of in the client's browser
8. A web server that can be installed for web testing
9. Windows servers for Apache, MySQL, and PHP
10. localhost
11. w3schools.com
12. Computers, operating systems, display, and browsers
13. Browsers
14. Chrome, Firefox, and Internet Explorer
15. The majority of screens are now high resolution.

Chapter 2: Building Web Pages with HTML

1. Hypertext Markup Language, HTML5
2. World Wide Web Consortium (W3C)
3. The way elements are formed, appear, and arranged in a well-formed script
4. HTML5 is not case sensitive, but it is recommended that you use lowercase.
5. Tags, which are shown with angle brackets (<>)
6. Attributes and their values. Attributes are shown as lowercase words that are related to their values with an equal sign, and values are placed in full double quotation marks.

7. Basic tags, which should have `<!DOCTYPE html>`, `<html></html>`, `<head></head>`, and `<body></body>`
8. Virtually any HTML tag
9. Any four of `<h2></h2>`, `<p></p>`, `<hr>`, ``, ` `, ``, `<!-- -->`, `<pre></pre>`, `<dl></dl>`, `<dt></dt>`, or `<dd></dd>`
10. No, only the `<pre></pre>` tag will repeat line endings and space within it.
11. Physical and logical character styles, where physical styles have strict definitions, and logical styles are defined by the browser.
12. No, you must replace those characters with the escape sequences `<` or `<`, `>` or `>`, `&` or `&` as button, checkbox, date, datetime, email, file, hidden, image, month, number, password, radio, reset, search, submit, tel, text, time, or week.
13. Including height and width speeds up loading, allowing text to be loaded while an image is drawn.
14. Hyperlinks provide the ability to click an object and transfer what is displayed by the browser (the *focus*) to an address associated with the object.
15. Any four of button, checkbox, date, datetime, email, file, hidden, image, month, number, password, radio, reset, search, submit, tel, text, time, or week
16. Any three of `<table></table>`, `<caption></caption>`, `<tr></tr>`, `<th></th>`, or `<td ></td>`
17. Handling of audio and video, date and time form input, and page sectioning
18. No, you must do that with styles.
19. Yes, you can use them. In the short term, they will work fine; over a longer term, browser support of them will diminish.
20. W3.org is the website for the World Wide Web Consortium, the committee that establishes the standards for what constitutes HTML.

Chapter 3: Styling with CSS

1. Cascading Style Sheets. CSS is a collection of rules for specifying the colors, fonts, and layout of a web page.
2. Embedded in the HTML file using the `<style>` tag; linked as an external CSS file using the `<link>` tag; and applied directly to an HTML tag using the `style` attribute

3. A CSS rule in its simplest form is a line of text with four components: a selector, a property, a value, and a declaration.
4. Selector is a tag, such as `<body>`; property is a feature of the tag, such as `background`; and value is an aspect of the feature, such as `yellow`. Declaration is the property/value pair.
5. The property and value are combined with a colon separating them, and a declaration is enclosed in curly braces.
6. With class names and IDs. These follow the selector with a period in front of a class name or a pound sign in front of an ID.
7. Cascading and inheritance; box and visual models of page layout; element positioning; units of measure; use of color, text and fonts, tables and lists
8. Through inheritance, an element's value can be determined by the values of elements that are earlier in the document tree.
9. The application of fonts and font characteristics, and the alignment and spacing of text
10. Choose common fonts, add alternative fonts, and add a generic font.
11. `<div>`, `<article>`, `<aside>`, `<footer>`, `<header>`, `<nav>`, `<section>`, `<h1>`, `<p>`, `<table>`
12. Position an element on a page, determine how elements interact with each other, the amount of separation between elements, how lists and tables are formed, and the characteristics and behavior of a page.
13. It is the visualization of a web page as a series of boxes within boxes, next to other boxes, and above and below still other boxes as a way of laying out a page.
14. Padding, the area separating the element from the border, the border itself, and margin, the area separating the border from either the page edge or the outer box of another element
15. Positioning determines how element boxes are sized and aligned relative to each other. Positioning methods are normal flow, the default method; relative positioning uses values for `top`, `right`, `bottom`, and `left` properties; float positioning is offset by the presence and characteristics of other elements; absolute positioning uses the values for `top`, `right`, `bottom`, and `left` properties; and with fixed positioning, which is the same as absolute, the element remains fixed as the screen is scrolled.
16. Float positioning allows you to pull a block box out of its normal-flow vertical position and attach it to the left or right side of the containing block. The benefit of a floated block is that it is flexible and can look good on many different viewing devices.

17. It limits the flexibility of a web page across various size displays and even various browsers, as well as adds difficulty in maintaining the page.
18. Decimals, decimals with leading zeroes, lowercase roman numerals, uppercase roman numerals, lowercase alphabet, uppercase alphabet, and lowercase Greek alphabet

Chapter 4: Adding Dynamic Elements with JavaScript

1. A programming language
2. False, it is interpreted.
3. True
4. True
5. False, JavaScript can also be contained within a separate .js page.
6. In the `<head>` and `<body>` sections
7. `<script></script>`
8. Using `//` to start a single line or by using `/* */` to enclose one or more lines of comments
9. While it isn't mandatory, good form dictates that you use only lowercase.
10. In JavaScript, you should end statements with a semicolon.
11. A series of characters (one of 256 letters, numbers, and special characters) enclosed in either single or double quotation marks
12. Arithmetic such as `+` `-` `*`, assignment such as `=` `+=`, bitwise such as `&` `-`, comparison such as `==` `!=`, logical such as `&&` `||`, and others such as `?:`
13. A *statement* is a line of code that is complete and correct in its syntax to perform a task. *Expressions* are anything that has a value. *Values* are anything that can be assigned to a variable.
14. A function is a piece of script that does something and can be repeatedly called within a larger script.
15. Objects can be the visual elements of a web page, such as windows, buttons, check boxes, and dates, and they can also be more abstract elements, such as math calculations and arrays.
16. Properties allow you to describe an object, distinguishing different variations of the same object from one another.
17. Methods are actions that objects can perform.

18. Event handlers recognize an event is taking place and then perform one or more tasks, adding interactivity between the web page and the user.
19. `if/else` statements; `while`, `do-while` statements; `for` statements; and `switch` statements
20. Focus is used to describe which window, or which element in a window, has the user's attention.
21. An *array* is a JavaScript object that allows you to store similar variables, indexed by number, so they can be used later in your script.
22. Button, check box, fieldset, hidden field, password, radio or option button, reset button, select button, submit button, text area, and text box
23. False, forms are created with HTML.
24. Forms are identified by name and by the `length` property.
25. Validation is a way of checking for correctness of user input in a form. Validation can check for the absence or presence of field content, if two items are equal or not, if particular characters are in a field, if a field is of a certain length, and if the mathematical product of one or more fields is a particular amount.

Chapter 5: Fundamentals of PHP

1. False. PHP operates in a server to provide server-side programming.
2. True, it is converted to machine language each time it is run.
3. An integrated developmental environment (IDE) aimed at PHP, such as Zend Studio; a developmental server with the latest version of PHP, such as Zend Server; and several of the latest browsers, such as Internet Explorer, Firefox, and Chrome
4. Change the file extension to `.php`, enclose PHP script in `<?php ... ?>`, and add PHP code.
5. False, it can be anywhere in an HTML file.
6. `echo()` and `print()`
7. Either single or double quote marks
8. Start a single comment line with either `#` or `//`, or enclose multiple lines with `/* */`.
9. End PHP statements with a semicolon (`;`), enclose strings in either single or double quote marks, separate arguments with commas (`,`), enclose function arguments in parentheses
10. The backslash character begins escape sequences, which are used to display literal characters such as a quote mark (`\"`) or a dollar sign (`\$`).

11. Arrays, hold multiple pieces of data; Booleans, either `TRUE` or `FALSE`; Floating point numbers, numbers with a fractional amount; Integers, whole numbers; `NULL`, the absence of a value; Strings, a set of characters; Objects, repeated script; Resources, external elements
12. The names of variables must begin with a `$`.
13. The names of labels must begin with either a letter `a–z` or `A–Z` or an underscore; can be of any length; and can contain letters, numbers, and underscores, but generally not special characters.
14. `=` means assign, as you might a value to a variable (`$a = 3` assigns the value 3 to `$a`); `==` checks for equality and returns `TRUE` if equal (if `$b = 3.0`, then `$a == $b` would have the value 1 or `TRUE`); `===` checks for identicalness and returns `TRUE` if two items are identical (`$a === $b` would have the value 0 or `FALSE` because 3 and 3.0 are not identical).
15. `++` means to increment by 1. `++$a` adds 1 to `$a` and then returns `$a`, whereas `$a++` returns `$a` and then adds 1 to it.
16. The order in which several mathematical or logical expressions are calculated. This is important in a long mathematical expression with several operators.
17. A statement is anything that is in between semicolons or the opening and closing PHP tags. Expressions are anything that has a value or evaluates to a value. Values are anything that can be assigned to a variable. Statements can contain expressions, and expressions can contain values and are values themselves.
18. A function is a piece of script that does something and can be repeatedly called within a larger script. Some examples of internal types of functions are array functions, date/time functions, math functions, and string functions.
19. Sunday, February 16th, 2014
20. Place the function above or before the `<!DOCTYPE html>` start of the HTML, and it won't be run until it is called.

Chapter 6: PHP Control and File Handling

1. Control structures allow you to switch execution in a script from one path to another based on some condition.
2. They allow you to specify that `if` some expression is `TRUE`, then a group of statements will be executed, `else` a different group of statements will be executed.

3. True. The `else` group of statements is optional and is needed only if you want to do something other than continue with the script if the conditional expression is `FALSE`.
4. False, you can test for “1” or “0.”
5. `?:` is the ternary operator and performs the same function as `if else`.
6. `while` tests the conditional expression before executing its statements, and `do-while` tests after execution.
7. `for` is used to loop through a piece of code, and in its basic form, the `for` expression is `for ($i = 1; $i <= 5; $i++)`, where `$i++` increments `$i` after it is used.
8. The `foreach` statement is used to iterate through arrays.
9. The `switch` statement is used when you want to compare a single variable to a number of different values and do something different depending on the value.
10. False. PHP’s file functions allow you to read and write files on the server.
11. `fopen()` establishes a connection or creates a file on the server and returns a file pointer.
12. The mode is used to indicate what will be done with the file and whether the file pointer is to the beginning or end of the file.
13. Cookies are placed and read by a PHP script and stored on the client until they expire to help identify the user.
14. Session variables are placed and read by a PHP script and stored on the server for the duration of a session to pass information among the pages of a website.
15. Server variables are created by a web server and read by a PHP script to provide information about the server.
16. Its stated purpose is to find the position of the first occurrence of one string in another string. It is most frequently used, though, to simply find if one string exists in another.

Chapter 7: PHP Arrays and Forms

1. *Arrays* are a means of grouping and handling several values with a single entity.
2. Values and keys
3. False. They can also use unique strings for keys.
4. False. You can add, replace, and delete values within an array.

5. Displays the `key=>value` pairs in an array
6. A looping function structured for arrays that successively places each of an array's values or `key=>value` pairs in variables that can be used and displayed
7. Values in ascending order, values in descending order, keys in ascending order, keys in descending order, values with their keys in ascending order, values with their keys in descending order
8. PHP uses a pointer to identify `key=>value` pairs, which can be moved within an array to return the value or key it is currently pointing to.
9. Convert arrays to and from strings, join arrays, split arrays, and rearrange arrays.
10. Show values in the first array, but not in the second; show `key=>value` pairs in the first array, but not in the second; show values in the first array that are also in the second; and show `key=>value` pairs in the first array that are also in the second
11. An array within an array
12. `<form>` `</form>` tag and the `<input />` tag
13. Placing the focus (cursor) in a field, and validating data that is entered
14. The `<form>` methods `get` and `post`
15. `$_GET`, `$_POST`, and `$_REQUEST`

Chapter 8: Putting PHP to Use

1. `isset()`
2. `header("Location:... ")` This function *must* be executed before any other output is sent to the user, such as an `echo` statement or the `<html>` or `<head>` elements.
3. `$_POST` or `$_GET` superglobal arrays
4. A CSS
5. With the `sha1()` function
6. Encrypt the password in the same way as originally done and then use the `strcmp()` function to see if it is in the data previously stored.
7. As the first line of a site page, insert `<?php require_once(enterSite.php); ?>`. Until this executes, the rest of the code in the script will not execute, unless `require` is replaced with `include`. The `_once` prevents the code from being loaded more than once.

Chapter 9: Introduction to Relational Databases

1. An organized way of storing information
2. Databases are made up of tables with rows or records and columns or fields.
3. A relational database places repetitive information into separate tables that are related to each other.
4. Two tables in a relational database are related to one another through their *key*, or index, a unique value assigned to each record in the table. In the table that it indexes, the key is called the *primary key*. When that key is used in another table, it is called a *foreign key*.
5. phpMyAdmin is an online graphical user interface for working with a MySQL database.
6. False. phpMyAdmin is meant to be used to initially create a MySQL database and to do occasional maintenance.
7. Open a browser and in the address bar type **localhost/phpMyAdmin**.
8. True. You must select the Null attribute to allow a field to be left blank.
9. By making the field type INT or integer and setting the auto-increment attribute
10. False. The collation sequence `latin1_swedish_ci` specifies how information is sorted and works fine with English and other Western European languages.

Chapter 10: Fundamentals of MySQL and SQL

1. phpMyAdmin, MySQL command line, and MySQL Workbench
2. SQL, or the Structured Query Language, is an efficient means of working with data using statements similar to spoken English. It is the foundation language used in MySQL.
3. SQL and MySQL commands and keywords should be in all uppercase, but this is not necessary.
4. Parts of the MySQL language include
 - Commands, which are declarative words such as `CREATE` and `SELECT`
 - Keywords, which are command-supporting words such as `FROM` and `WHERE`
 - Names, which identify elements such as databases and tables
 - Literals, which are used as values in a statement such as numbers and dates
 - User variables, which are names that can be assigned values and begin with an `(@)` sign

- Expressions, which are words or phrases that provide values in a MySQL statement
 - Reserved words for which MySQL has a special usage
 - Comments, which can begin with a double dash, a hash mark, or a slash mark and asterisk
 - Escape sequences, which begin with a backslash
5. Literals include string values, numeric values, and date and time values.
 6. Dates can be strings with delimiters ('YYYY-MM-DD'), strings without delimiters ('YYYYMMDD'), and numbers (YYYYMMDD).
 7. Arithmetic operators such as +, *, and %; comparison operators such as =, >, and !=; logical operators such as &&, !, and ||; assignment operators such as :=
 8. Functions include
 - Arithmetic functions, such as AVG(), COUNT(), and RAND()
 - Comparison functions, such as GREATEST(), IN(), and STRCMP()
 - Control flow functions, such as CASE(), IF(), and NULLIF()
 - Date and time functions, such as ADDDATE(), DATE(), and NOW()
 - Encryption and compression functions, such as AES_ENCRYPT(), COMPRESS(), and ENCODE()
 - String functions, such as CONCAT(), FORMAT(), and TRIM()
 - System and information functions, such as DATABASE(), USER(), and VALUES()
 9. ROUND considers the fractional part of a number, so 4.6 is rounded to 5, while TRUNCATE simply cuts off the fractional part, so 4.6 is truncated to 4.
 10. MySQL wildcard characters are %, which can replace any number of characters, and _, which can replace a single character.
 11. %M %D, %Y
 12. VARBINARY or BLOB binary string data types

Chapter 11: Implementing MySQL Command Statements

1. Data definition statements such as ALTER and CREATE, data manipulation statements such as INSERT and SELECT, and other statements such as EXECUTE and SHOW
2. Commands, which define MySQL statements and keywords, functions, and operators in clauses that modify the command

3. An asterisk (*) indicates that all columns or fields are referenced.
4. A literal is an object that is to be taken at its face value—for example, a number.
5. Put commands and keywords in all capital letters; put database, table, column, and other object names in lowercase; and end the statement with a semicolon (;).
6. Table name, column name, column type and size, and column characteristics
7. `AUTO_INCREMENT` automatically increments an integer or floating point data type by 1. It is generally used when defining an automatically inserted key or index.
8. A primary key is the primary index for the table in which it exists. A foreign key is the primary key for a table other than the one it is in.
9. The `TIMESTAMP` data type is used to hold a date and time value. The `TIMESTAMP` data type automatically adds the current date and time value when a table record is inserted and then updates the data and time value when the record is updated.
10. `DECIMAL(10, 2)`, which means that it is decimal number with ten digits and two to the right of the decimal point.
11. `USE` the database.
12. False. You can insert a number of sets of values that become records in the table.
13. Retrieve information from the table.
14. `SELECT column_names FROM table_name WHERE conditional_expression ORDER BY column_name;`
15. Counts the number of rows that are returned in the `SELECT`
16. The `REPLACE` command and statement
17. `REPLACE` uses the primary key, if it exists, to locate and change a record in a table, or, if it doesn't exist, to insert a new record. `UPDATE` can use any column in a table to locate a record and change its column contents. `UPDATE` isn't used to insert a new record.
18. False. `DELETE` removes records or rows from a table.
19. The `ALTER` command and statement can be used to add columns, change a column default, change a column name and definition, drop a column, and rename a table.
20. Stored programs are MySQL statements and keywords that are stored with a database and executed at a later time. Events, views, and triggers are examples of stored programs.

21. Turn on the Event Scheduler.
22. Events are the scheduled execution of MySQL statements.
23. At a particular date and time, on a periodic interval, on or within a start and end date and time
24. Views are, in essence, named `SELECT` statements that you can repeatedly call using only the name.
25. Triggers are one or more statements that are executed on each row of a particular table when an event occurs within the table.

Chapter 12: Using a MySQL Database with PHP

1. PHP uses a set of functions that perform SQL commands.
2. False. You should use only `mysqli` functions. The `mysql` functions have been deprecated.
3. False. It is commonly tested to see if the connection was made.
4. The connection ID, which is used by other functions
5. It is used to test whether MySQL PHP functions are successful and, if not, an error message can be generated and the script terminated. It should only be used in testing because the user is not given any alternative to restarting the script.
6.

```
if(!result) { echo "message" . mysqli_error(connect ID) };
```
7. False. It is a good practice, but it is optional.
8. The database MySQL information functions
9. The `mysqli_insert_id()` function will provide it.
10. The `mysqli_ping` function will check the connection and reconnect if necessary.
11. True. It is a complex partnership.
12. Superglobal arrays. The two most common form-related examples are `$_GET` and `$_POST`.
13. The user clicks a Submit button causing the `$_POST['submit']` element to be `TRUE`.
14. False. If there is no `action` attribute in the `<form` tag, it will be assumed that the PHP code is in the same script.
15. `mysqli_real_escape_string()`

Chapter 13: Registering and Responding to Users

1. The steps needed for web app creation are
 - a. Identify what the app will do.
 - b. Specify the needed components in terms of database tables and scripts.
 - c. Design how the screens or pages will look in terms of layout, fonts, colors, and lines.
 - d. Create a page template in HTML for the desired look.
 - e. Create a CSS for the desired look.
 - f. Identify the necessary fields and characteristics and then create the needed database tables.
 - g. Create a detailed flowchart of the app to identify the functions that must be in each script.
 - h. Write, test, and debug the scripts.
2. Copying large amounts of code from script to script
3. Heading, horizontal navigation bar, vertical navigation bar, main content, and footer
4. With the PHP `include()`, `include_once()`, `require()`, and `require_once()` functions
5. Transferring through the URL with POST and GET, and transferring creating elements in the `$_SESSION` variable
6. PHP and HTML can be combined in a single script with the PHP referred to through its variables and `$_SERVER['PHP_SELF']` or by having separate scripts where the two scripts call each other and transfer information.
7. User errors can be identified and flagged with JavaScript or by identifying them with PHP, transferring them in a `$_SESSION` variable, and flagging them in the HTML.
8. JavaScript is used here to place the cursor in a starting field and to check and flag errors.
9. `session_start();`
10. With quotation marks
11. False. You must separately compare what was found to what you were looking for to determine the outcome.
12. Add a zero (+0) to the string or casting with (int) or (float).

Chapter 14: Handling Online Purchases

1. Modify an existing app.
2. Store only a link to the image in the database, and store the images in a separate folder within the folder with the web pages.
3. Place the non-numeric value in single quotes.
4. PHP will interpret the value name or variable as a literal value.
5. In PHP, it is `$email`; in JavaScript, it is `email`; in MySQL, it is `'$email'`; and in HTML, it is `<?php echo $email; ?>`.
6. By ordering the table by the index in descending order with the statement:

```
$query = "SELECT * FROM table_name ORDER BY index_name DESC";
```
7. Use `mysqli_insert_id` to pick up an automatically generated index.
8. Place a period (.) between the elements.