

Chapter 10 Answers

1. HTTP was originally designed to store data about individual visits to a Web site. True or False?
2. Stored information about a previous visit to a Web site is called _____ information.
 - a. HTTP
 - b. client-side
 - c. state
 - d. prior

3. Describe the different types of information about a user that a Web server might need to store.

- Customize individual Web pages based on user preferences.
- Temporarily store information for a user as a browser navigates within a multipart form.
- Allow a user to create bookmarks for returning to specific locations within a Web site.
- Provide shopping carts that store order information.
- Store user IDs and passwords.
- Use counters to keep track of how many times a user has visited a site.

4. Explain how to use form fields to temporarily store user information.

You create hidden form fields with the `<input>` element. Hidden form fields temporarily store data that needs to be sent to a server along with the rest of a form, but that a user does not need to see. Examples of data stored in hidden fields include the result of a calculation or some other type of information that a program on the Web server might need. You create hidden form fields using the same syntax used for other fields created with the `<input>` element: `<input type="hidden">`. The only attributes that you can include with a hidden form field are the `name` and `value` attributes.

5. In what format are items in a query string appended to a target URL?
 - a. in comma-delimited format
 - b. as name&value pairs
 - c. as name=value pairs
 - d. in name, value, length format

6. Explain how to use PHP to access query string data that is appended to a URL.

You can access any query string data that is appended to a URL from PHP by using the `$_GET[]` autoglobal, the same as for any forms that are submitted with the GET method.

7. What is the correct syntax for creating a temporary cookie containing a value of "blue"?

a. `$Color = setcookie("blue");`

b. `setcookie("color", "blue");`

c. `setcookie("blue", "color");`

d. `setcookie("blue");`

8. You must manually encode and decode cookie values. True or False?

9. By default, cookies created without the `expires` argument of the `setcookie()` function are available for 24 hours. True or False?

10. Cookies created without the `expires` argument of the `setcookie()` function are called _____.

a. transient

b. temporary

c. permanent

d. persistent

11. Which of the following examples specifies that a cookie should expire in three days?

a. `time()+48h`

b. `time()+24h*3`

c. `time()+60*60*24*7`

d. `time()+60*60*24*3`

12. The availability of a cookie to other Web pages on a server is determined by the _____ argument of the `setcookie()` function.

a. path

b. directory

c. system

d. server

13. Which argument of the `setcookie()` function is used for sharing cookies outside of a domain?

- a. `domain`
- b. `share`
- c. `secure`
- d. You cannot share cookies outside of a domain.

14. You use the _____ to read cookies in PHP.

- a. `$_COOKIE` autoglobal
- b. `$_COOKIES` autoglobal
- c. `cookie()` function
- d. `getcookie()` function

15. How do you delete cookies before the time assigned to the `setcookie()` function's expires argument elapses?

- a. Assign a NULL value with the `setcookie()` function.
- b. Subtract any number of seconds from the `time()` function.
- c. Execute the `deletecookie()` function.
- d. You cannot delete a cookie before the time assigned to the `setcookie()` function's expires argument elapses.

16. Explain the security risks involved with cookies and how sessions offer a more secure method of maintaining state.

Many clients do not accept cookies due to the rampant rise of spyware, which is an annoying type of software that gathers user information from a local computer for marketing and advertising purposes, but without the user's knowledge. Users are increasingly choosing to disable cookies to prevent spyware from gathering user information from stored cookies.

17. Unlike the `setcookie()` function, you can call the `session_start()` function from any location on a Web page. True or False?

18. What is the name of the cookie that PHP creates for a session?

- a. `SESSION`
- b. `PHPSESSION`
- c. `SESSIONID`

d. PHPSESSID

19. Explain how to pass a session ID to other PHP scripts when cookies are not available.

If a client's Web browser is configured to accept cookies, the session ID is assigned to a temporary cookie named PHPSESSID. However, because you cannot be certain that every client accepts cookies, you should also pass the session ID as a query string or hidden form field to any Web pages that are called as part of the current session. You pass a session ID in a name=value pair of PHPSESSID=session ID. You use the `session_id()` function to retrieve the session ID for the current session. You can also use the constant SID, which contains a string consisting of "PHPSESSID=" and the session ID. For hidden form fields, assign a value of "PHPSESSID" to the name attribute and use the `session_id()` function to assign the session ID to the value attribute of the `<input>` element.

20. You use the _____ to access session variables in PHP.

a. `$_SESSION` autoglobal

b. `$_SESSIONS` autoglobal

c. `session()` function

d. `getsession()` function

Chapter 11 Answers

1. Reusable software objects are often referred to as _____.

a. methods

b. components

c. widgets

d. functions

2. Explain the benefits of object-oriented programming.

The term object-oriented programming (OOP) refers to the creation of reusable software objects that can be easily incorporated into multiple programs. The term object specifically refers to programming code and data that can be treated as an individual unit or component. (Objects are often also called components.) For example, you might create a `Loan` object that calculates the number of payments required to pay off a loan. The `Loan` object might also store information such as the principal loan amount and the interest rate. The term data refers to information contained within variables or other types of storage structures. The functions associated with an object are called methods, and the variables that are associated with an object are called properties or attributes. In the `Loan` object example, a function that calculates the number of payments required to pay off the loan is a method. The principal loan amount and the interest rate are properties of the `Loan` object.

Popular object-oriented programming languages include C++, Java, and Visual Basic. Using any of these or other object-oriented languages, programmers can create objects themselves or use objects created by other programmers. For example, if you are creating an accounting program in Visual Basic, you can use an object named `Payroll` that was created in C++. The `Payroll` object might contain one method that calculates the amount of federal and state tax to deduct, another function that calculates the FICA amount to deduct, and so on. Properties of the `Payroll` object might include an employee's number of tax withholding allowances, federal and state tax percentages, and the cost of insurance premiums. You do not need to know how the `Payroll` object was created in C++, nor do you need to re-create it in Visual Basic. You only need to know how to access the methods and properties of the `Payroll` object from the Visual Basic program.

3. The functions associated with an object are called _____.
 - a. properties
 - b. fields
 - c. methods
 - d. attributes
4. The term "black box" refers to _____.
 - a. a property
 - b. debugging
 - c. encapsulation
 - d. an interface
5. A(n) _____ is an object that has been created from an existing class.
 - a. pattern
 - b. structure
 - c. replica
 - d. instance
6. An object inherits its characteristics from a class. **True** or False?
7. A function that is used as the basis for an object is called a(n) _____.
 - a. method
 - b. class
 - c. class constructor
 - d. object variable

8. Which of the following characters are used in member selection notation?

- a. >
- b. ->
- c. =>
- d. .

9. What is the correct syntax to connect to the MySQL database server using object-oriented style?

- a. `$Variable = mysqli_connect("localhost", "user", "password", "database_name");`
- b. `$Variable = new mysqli_connect("localhost", "user", "password", "database_name");`
- c. `$Variable = mysqli("localhost", "user", "password", "database_name");`
- d. `$Variable = new mysqli("localhost", "user", "password", "database_name");`

10. Explain how to handle MySQL errors using object-oriented database style.

When you use procedural syntax to connect to the MySQL database server, the `mysqli_connect()` function returns a value of false if the database connection attempt fails. However, when you use the `mysqli()` constructor function to instantiate a new database object from the `mysqli` class, an object is instantiated even if the database connection fails. For this reason, with object-oriented style, you cannot simply terminate script execution with the `die()` or `exit()` functions in the event that the database connection fails, as you do with procedural syntax. With object-oriented style, you must check whether a value is assigned to the `mysqli_connect_errno()` or `mysqli_connect_error()` functions and then call the `die()` function to terminate script execution.

11. The terms variable and object are often used interchangeably in object-oriented programming. True or False?

12. Class names usually begin with a(n) _____ to distinguish them from other identifiers.

- a. number
- b. exclamation mark (!)
- c. ampersand (&)
- d. uppercase letter

13. Which of the following functions returns the name of the class upon which an object is based?

- a. `class_of()`
- b. `instanceof()`
- c. `class_name()`
- d. `get_class()`

14. What extensions should you use for external HTML files and PHP scripts and why?

You can use any file extension you want for include files. Many programmers use an extension of `.inc` for HTML and other types of information that do not need to be processed by the Web server. Although you can use the `.inc` extension for external files containing PHP scripts, you should avoid doing so unless your Web server is configured to process `.inc` files as PHP scripts. If your Web server is not configured to process `.inc` files as PHP scripts, anyone can view the contents of the file simply by entering the full URL in a Web browser. This creates a potential security risk, especially if the external file contains proprietary code or sensitive information such as passwords. Because most Web servers process the contents of a PHP script and only return HTML to the client, your safest bet is to always use an extension of `.php` for external files that contain PHP code.

15. Explain the principle of information hiding.

One of the fundamental principles in object-oriented programming is the concept of information hiding. Information hiding gives an encapsulated object its black box capabilities so that users of a class can see only the members of the class that you allow them to see. Essentially, the principle of information hiding states that any class members that other programmers, sometimes called clients, do not need to access or know about should be hidden. Information hiding helps minimize the amount of information that needs to pass in and out of an object, which helps increase program speed and efficiency. Information hiding also reduces the complexity of the code that clients see, allowing them to concentrate on the task of integrating an object into their programs.

16. Which of the following access specifiers prevents clients from calling member functions or accessing data members?

- a. `public`
- b. `private`
- c. `protected`
- d. `privileged`

17. You name a constructor function as the same name as its class or as _____.

- a. `construct()`
- b. `__construct()`
- c. `constructor()`
- d. `__constructor()`

18. When is a destructor called? (Choose all that apply.)

- a. when a script ends
- b. when the constructor function ends
- c. when you delete a class object with the `unset()` function
- d. when you call the `serialize()` function

19. Explain the use of accessor functions. How are accessor functions often named?

Even if you make all data members in a class private, you can still allow your program's clients to retrieve or modify the value of data members via accessor functions. Accessor functions are public member functions that a client can call to retrieve or modify the value of a data member. (You learn more about how to use access specifiers with member functions later in this chapter.) Because accessor functions often begin with the words "set" or "get," they are also referred to as set or get functions. Set functions modify data member values; get functions retrieve data member values. To allow a client to pass a value to your program that will be assigned to a private data member, you include parameters in a set function's definition. You can then write code in the body of the set function that validates the data passed from the client, prior to assigning values to private data members. For example, if you write a class named `Payroll` that includes a private data member containing the current state income-tax rate, you could write a public accessor function named `getStateTaxRate()` that allows clients to retrieve the variable's value. Similarly, you could write a `setStateTaxRate()` function that performs various types of validation on the data passed from the client (such as making sure the value is not null, is not greater than 100%, and so on) prior to assigning a value to the private state tax rate data member.

20. When serializing objects, how do you specify which data members to serialize?

You don't necessarily have to serialize each and every data member in a class, particularly for large objects that contain numerous data members. Instead, you could use a `__sleep()` function to specify which data members to serialize. If you do include a `__sleep()` function in your class, the function must return an array of the data members to serialize or you will receive an error.

Chapter 12 Answers

1. If the PHP scripting engine encounters a problem while a program is executing, that problem is called a(n) _____ error.
 - a. application
 - b. logic
 - c. run-time
 - d. syntax
2. PHP error messages help you find logic errors in your programs. **True** or False?
3. _____ errors are problems in the design of a program that prevent it from running as you anticipate.

- a. Application
 - b. Logic**
 - c. Run-time
 - d. Syntax
4. _____ errors occur when you enter code that the scripting engine does not recognize.
- a. Application
 - b. Logic
 - c. Run-time
 - d. Syntax**
5. Which of the following types of error messages occur when a script contains a run-time error that prevents it from executing?
- a. parse error messages
 - b. fatal error messages**
 - c. warning messages
 - d. notice messages
6. Error messages point to the exact location in a script that is causing the error. True or **False**?
7. Which of the following statements causes a syntax error?
- a. `echo "<p>Hello World</p>";`
 - b. `print("<p>Hello World</p>")`**
 - c. `Return true;`
 - d. `$Clients = array();`
8. Which of the following functions causes a run-time error?
- a.

```
function calcMarginPercent() {  
    $GrossProfit = 100;  
    $NetProfit = 100;  
    $Margin = $GrossProfit - $NetProfit;  
    $MarginPercent = $Margin / $GrossProfit;  
}
```
 - b.

```
function calcMarginPercent() {  
    $GrossProfit = 200;
```

```

    $NetProfit = 100;
    $Margin = $GrossProfit - $NetProfit;
    $MarginPercent = $Margin / $GrossProfit;
}

```

c. function calcMarginPercent() {
 \$GrossProfit = 200;
 \$NetProfit = 100;
 \$Margin = \$GrossProfit - \$NetProfit;
 \$MarginPercent = \$Margin / \$GrossProfit;
 }

d. function calcMarginPercent() {
 \$GrossProfit = 0;
 \$NetProfit = 100;
 \$Margin = \$GrossProfit - \$NetProfit;
 \$MarginPercent = \$Margin / \$GrossProfit;
 }

9. Which of the following if statements is logically incorrect?

a. if (\$Count < 5)
 echo "<p>\$Count</p>";

b. if (\$Count =< 5)
 echo "<p>\$Count</p>";

c. if (\$Count = 5);
 echo "<p>\$Count</p>";

d. if (\$Count = 5) {
 echo "<p>\$Count</p>";
 }

10. Explain why the PHP Group strongly recommends that you turn off the display of errors for scripts that run in production environments.

The php.ini configuration file contains two directives, display_errors and display_startup_errors, that determine whether error messages print to a Web browser. The display_errors directive prints script error messages, whereas the display_startup_errors directive displays errors that occur when PHP first starts. By default, the display_errors directive is assigned a value of "On," and the display_startup_errors directive is assigned a value of "Off." Although displaying error messages is useful when you develop PHP scripts, the PHP Group strongly recommends that you turn this feature off for scripts that run in production environments and instead save any errors in a log file. The reason for this is because hackers can use any displayed error messages to identify potential weaknesses in your Web site. The PHP Group also recommends that you only turn the display_startup_errors directive on when debugging a script.

11. Which of the following error reporting levels report run-time errors? (Choose all that apply.)

a. `E_ERROR`

b. `E_WARNING`

c. `E_PARSE`

d. `E_NOTICE`

12. To which of the following directives do you assign the path and name of the file where PHP will log errors?

a. `log_errors`

b. `error_log`

c. `logging`

d. `log_location`

13. Which of the following statements correctly specifies a custom error-handling function named `logProblems()`?

a. `logProblems = set_error_handler();`

b. `set_error_handler = logProblems();`

c. `set_error_handler("logProblems");`

d. `set_error_handler(logProblems());`

14. When you assign a custom function to handle errors, PHP's default error-handling functionality is completely bypassed for all types of error reporting levels. True or False?

15. Which of the following functions logs an error message to the location specified by the `error_log` directive in the `php.ini` configuration file?

a. `error_log()`

b. `log_error()`

c. `log()`

d. `error()`

16. Which of the following error reporting levels can you pass as the second argument of the `trigger_error()` function? (Choose all that apply.)

a. `E_ALL`

b. `E_USER_ERROR`

Comment [KA1]: AU: What preceding types? Are these listed in a previous question? If so, please state the question number so the reader knows where to look to see the preceding types. Thanks, KA

c. E_USER_WARNING

d. E_USER_NOTICE

17. Which of the following functions allow you to print a color highlighted version of a file to a Web browser?

a. color_file()

b. file_color()

c. highlight_file()

d. file_highlight()

18. _____ refers to the examination of individual statements in an executing program.

a. Trailing

b. Tracing

c. Tracking

d. Commenting

19. Explain how to use a driver program.

When using echo() statements to trace bugs, it is helpful to use a **driver program**, which is a simplified, temporary program that is used for testing functions and other code. A driver program is simply a PHP program that contains only the code you are testing. Driver programs do not have to be elaborate; they can be as simple as a single function you are testing. This technique allows you to isolate and test an individual function without having to worry about Web page elements, event handlers, global variables, and other code that form your program's functionality as a whole.

20. Which of the following code structures prints the text "Hello World" five times?

```
a. for ($Count = 1; $Count < 6; ++$Count) ;
    echo "<p>Hello World</p>";
    echo "<p>Hello World</p>";
    echo "<p>Hello World</p>";
    echo "<p>Hello World</p>";
    echo "<p>Hello World</p>";
```

```
b. for ($Count = 0; $Count < 6; ++$Count)
    echo "<p>Hello World</p>";
```

```
c. for ($Count = 0; $Count < 6; ++$Count) {
    echo "<p>Hello World</p>";
}
```

d. `for ($Count = 0; $Count < 6; ++$Count);
echo "<p>Hello World</p>";`

