**PHP Basics**

1. How to achieve multiple inheritance in PHP?

PHP doesn't have multiple inheritance. If you have PHP 5.4, though, you can use [traits](http://php.net/manual/en/language.oop5.traits.php) to at least avoid every class having to copy code.

1. interface A {
2. public function do1();
3. public function do2();
4. public function do3();
5. }
6. trait Alike {
7. public function do1() { }
8. public function do2() { }
9. public function do3() { }
10. }
11. interface B {
12. public function do4();
13. public function do5();
14. public function do6();
15. }
16. trait Blike {
17. public function do4() { }
18. public function do5() { }
19. public function do6() { }
20. }
21. class C implements A, B {
22. use Alike, Blike;
23. }
24. class D implements A {
25. use Alike;
26. // You can even "override" methods defined in a trait
27. public function do2() { }
28. }
29. 2. Difference between Zend framework 2 VS Zend framework 1?

3. What is curl?How to pass data in header while using CURL?

See <https://stackoverflow.com/questions/3062324/what-is-curl-in-php>

4. What are the different tables (Engine) present in MySQL, which one is the default?

5. Explain the difference between MyISAM Static and MyISAM Dynamic

6. How can I retrieve values from one database server and store them in other database server using PHP?

7. What is the difference between if (‘value’ == $var) and if ($var == ‘value’)?

There is no functional difference. But some people have a defensive programming habit to put constant values on the left of the comparison to avoid silent bugs if they mistype = instead of ==.  
  
For example, the following is valid code, but it causes a bug. The value is assigned to the variable, and if the value is nonzero, the condition is true.  
  
**if** ($var = 'value') ...   
  
If you put the constant on the left, you'll catch such mistakes at PHP parse time.  
  
**if** ('value' = $var) /\* ERROR! \*/

8. How can we prevent SQL-injection in PHP?

9. In how many ways we can retrieve the data in the result set of MySQL using PHP? What is the difference between mysql\_fetch\_object and mysql\_fetch\_array?

There are (more than) three popular ways to use MySQL from PHP.

1. (**DEPRECATED**) The [mysql functions](http://www.php.net/manual/en/ref.mysql.php) are procedural and use manual escaping.
2. [mysqli](http://www.php.net/manual/en/intro.mysqli.php) is a replacement for the mysql functions, with object-oriented and procedural versions. It has support for prepared statements.
3. [PDO](http://php.net/manual/en/book.pdo.php) (PHP Data Objects) is a general database abstraction layer with support for MySQL among many other databases. It provides prepared statements, and significant flexibility in how data is returned.

I would recommend using PDO with prepared statements. It is a well-designed API and will let you more easily move to another database (including any that supports ODBC) if necessary.

———————————————————

Those are differents API to access a MySQL backend

* the [mysql](http://php.net/manual/en/book.mysql.php) is the historical API
* the [mysqli](http://jp2.php.net/manual/en/book.mysqli.php) is a new version of the historical API, it should perform better and have a better set of function, also the API is object oriented
* [PDO\_MySQL](http://jp2.php.net/manual/en/ref.pdo-mysql.php), is the MySQL for PDO, PDO has been introduced in PHP, the project aims to make a common API for all the databases access, so in theory you should be able to migrate between RDMS without changing any code(if you don’t use specific RDBM function in your queries), also object oriented

10. After the code below is executed, what will be the value of $text and what will strlen($text) return?

$text = ‘Mike ‘;

$text[10] = ‘Sam';

11. Consider the following code:

$x = NULL;

if (‘0xFF’ == 255) {

$x = (int)’0xFF';

}

What will be the value of $x after this code executes? Explain your answer.

12. What are SQL Injections, how do you prevent them and what are the best practices?

13. Suppose that you have to implement a class named Dragonball. This class must have an attribute named ballCount (which starts from 0) and a method iFoundaBall. When iFoundaBall is called, ballCount is increased by one. If the value of ballCount is equal to seven, then the message My name is John is printed, and ballCount is reset to 0. How would you implement this class?

14. What’s the difference between using mysql\_ functions and PDO?

1. PDO has 12 different driver supports while MYSQLI has only one i.e MYSQL.
2. PDO uses OOP API while MySqli use both OOP and precedural
3. PDO has named parameters but MuSqli does not.
4. With PDO one can have prepared statements at client side

15. How can you enable error reporting in PHP?

16. How to use “MySQL in clause” for “string search” and “number search”?

17. Do you know what the PHP-FIG is? Describe it, describe the PSRs you know.

18. What are the reasons for selecting lamp (Linux, apache, MySQL, PHP) instead of the combination of other software programs, servers, and operating systems?

19. What will be the output of the code below and why?

$x = 5;

echo $x;

echo “&lt;br /&gt;”; // <br/>

echo $x+++$x++; // 11

echo “&lt;br /&gt;”; // <br/>

echo $x; // 7

echo “&lt;br /&gt;”; <br/>

echo $x—$x–; //1

echo “&lt;br /&gt;”; //<br/>

echo $x; // 5

20. What is wrong with this code?

class SomeClass

{

protected $\_someMember;

public function \_construct()

{

$this->\_someMember = 1;

}

public static function getSomethingStatic()

{

return $this->\_someMember \* 5 ;// here’s the catch

}

}

No Problem.

21. How does one prevent the following Warning ‘Warning: Cannot modify header information – headers already sent’ and why does it occur in the first place?

22. Are objects in PHP 5 passed by value or reference? Explain

23. Why would you use === instead of ==? Explain

24. What will be the output of each of the statements below and why?

var\_dump(0123 == 123);  
var\_dump(‘0123′ == 123);  
var\_dump(‘0123′ === 123);

25. How do you define a constant in PHP? How do you determine if a constant is defined in PHP?

26. When I open up a website you developed and it’s a blank white screen, what steps do you take to troubleshoot the problem?

27.  What classes would you create to build a basic Twitter-style status system with OOP?

28. Describe your experiences with web services.

29. Have you ever used a PHP framework? What frameworks are you experienced in?

30. What frameworks do you prefer? Why?

**What is the purpse $\_PHP\_SELF variable?**

The PHP default variable $\_PHP\_SELF is used for the PHP script name and when you click "submit" button then same PHP script will be called.

**What does PEAR stands for?**

PEAR means “PHP Extension and Application Repository”. it extends PHP and provides a higher level of programming for web developers.

**How can we display the output directly to the browser?**

To be able to display the output directly to the browser, we have to use the special tags <?= and ?>.

**10) What is the main difference between PHP 4 and PHP 5?**

PHP 5 presents many additional OOP (Object Oriented Programming) features.

**11) Is multiple inheritance supported in PHP?**

PHP includes only single inheritance, it means that a class can be extended from only one single class using the keyword ‘extended’.

**12) What is the meaning of a final class and a final method?**

‘final’ is introduced in PHP5. Final class means that this class cannot be extended and a final method cannot be overrided.

**13) How comparison of objects is done in PHP5?**

We use the operator ‘==’ to test is two object are instanced from the same class and have same attributes and equal values. We can test if two object are refering to the same instance of the same class by the use of the identity operator ‘===’.

**14) How can PHP and HTML interact?**

It is possible to generate HTML through PHP scripts, and it is possible to pass informations from HTML to PHP.

**15) What type of operation is needed when passing values through a form or an URL?**

If we would like to pass values througn a form or an URL then we need to encode and to decode them using htmlspecialchars() and urlencode().

**16) How can PHP and Javascript interact?**

PHP and Javascript cannot directly interacts since PHP is a server side language and Javascript is a client side language. However we can exchange variables since PHP is able to generate Javascript code to be executed by the browser and it is possible to pass specific variables back to PHP via the URL.

**17) What is needed to be able to use image function?**

GD library is needed to be able execute image functions.

**18) What is the use of the function ‘imagetypes()’?**

imagetypes() gives the image format and types supported by the current version of GD-PHP.

**19) What are the functions to be used to get the image’s properties (size, width and height)?**

The functions are getimagesize() for size, imagesx() for width and imagesy() for height.

**20) How failures in execution are handled with include() and require() functions?**

If the function require() cannot access to the file then it ends with a fatal error. However, the include() function gives a warning and the PHP script continues to execute.

**21) What is the main difference between require() and require\_once()?**

require() and require\_once() perform the same task except that the second function checks if the PHP script is already included or not before executing it.

(same for include\_once() and include())

**22) How can I display text with a PHP script?**

Two methods are possible:

|  |  |
| --- | --- |
| 1 | <!--?php echo "Method 1"; print "Method 2"; ?--> |

**23) How can we display information of a variable and readable by human with PHP?**

To be able to display a human-readable result we use print\_r().

**24) How is it possible to set an infinite execution time for PHP script?**

The set\_time\_limit(0) added at the beginning of a script sets to infinite the time of execution to not have the PHP error ‘maximum execution time exceeded’.It is also possible to specify this in the php.ini file.

**25) What does the PHP error ‘Parse error in PHP – unexpected T\_variable at line x’ means?**

This is a PHP syntax error expressing that a mistake at the line x stops parsing and executing the program.

**26) What should we do to be able to export data into an Excel file?**

The most common and used way is to get data into a format supported by Excel. For example, it is possible to write a .csv file, to choose for example comma as separator between fields and then to open the file with Excel.

**27) What is the function file\_get\_contents() usefull for?**

file\_get\_contents() lets reading a file and storing it in a string variable.

**28) How can we connect to a MySQL database from a PHP script?**

To be able to connect to a MySQL database, we must use mysql\_connect() function as follows:

|  |  |
| --- | --- |
| 1 | <!--?php $database = mysql\_connect("HOST", "USER\_NAME", "PASSWORD"); mysql\_select\_db("DATABASE\_NAME",$database); ?--> |

**29) What is the function mysql\_pconnect() usefull for?**

mysql\_pconnect() ensure a persistent connection to the database, it means that the connection do not close when the the PHP script ends.

**30) How the result set of Mysql be handled in PHP?**

The result set can be handled using mysql\_fetch\_array, mysql\_fetch\_assoc, mysql\_fetch\_object or mysql\_fetch\_row.

**31) How is it possible to know the number of rows returned in result set?**

The function mysql\_num\_rows() returns the number of rows in a result set.

**32) Which function gives us the number of affected entries by a query?**

mysql\_affected\_rows() return the number of entries affected by an SQL query.

**33) What is the difference between mysql\_fetch\_object() and mysql\_fetch\_array()?**

The mysql\_fetch\_object() function collects the first single matching record where mysql\_fetch\_array() collects all matching records from the table in an array.

**34) How can we access the data sent through the URL with the GET method?**

In order to access the data sent via the GET method, we you use $\_GET array like this:

www.url.com?var=value  
$variable = $\_GET[“var”]; this will now contain ‘value’

**35) How can we access the data sent through the URL with the POST method?**

To access the data sent this way, you use the $\_POST array.

Imagine you have a form field called ‘var’ on the form, when the user clicks submit to the post form, you can then access the value like this:

$\_POST[“var”];

**36) How can we check the value of a given variable is a number?**

It is possible to use the dedicated function, is\_numeric() to check whether it is a number or not.

**37) How can we check the value of a given variable is alphanumeric?**

It is possible to use the dedicated function, ctype\_alnum to check whether it is an alphanumeric value or not.

**38) How do I check if a given variable is empty?**

If we want to check whether a variable has a value or not, it is possible to use the empty() function.

**39) What does the unlink() function means?**

The unlink() function is dedicated for file system handling. It simply deletes the file given as entry.

**40) What does the unset() function means?**

The unset() function is dedicated for variable management. It will make a variable undefined.

**41) How do I escape data before storing it into the database?**

addslashes function enables us to escape data before storage into the database.

**42) How is it possible to remove escape characters from a string?**

The stripslashes function enables us to remove the escape characters before apostrophes in a string.

**43) How can we automatically escape incoming data?**

We have to enable the Magic quotes entry in the configuration file of PHP.

**44) What does the function get\_magic\_quotes\_gpc() means?**

The function get\_magic\_quotes\_gpc() tells us whether the magic quotes is switched on or no.

**45) Is it possible to remove the HTML tags from data?**

The strip\_tags() function enables us to clean a string from the HTML tags.

**46) what is the static variable in function useful for?**

A static variable is defined within a function only the first time and its value can be modified during function calls as follows:

|  |  |
| --- | --- |
| 1 | <!--?php function testFunction() { static $testVariable = 1; echo $testVariable; $testVariable++; } testFunction();        //1 testFunction();        //2 testFunction();        //3 ?--> |

**47) How can we define a variable accessible in functions of a PHP script?**

This feature is possible using the global keyword.

**48) How is it possible to return a value from a function?**

A function returns a value using the instruction ‘return $value;’.

**49) What is the most convenient hashing method to be used to hash passwords?**

It is preferable to use crypt() which natively supports several hashing algorithms or the function hash() which supports more variants than crypt() rather than using the common hashing algorithms such as md5, sha1 or sha256 because they are conceived to be fast. hence, hashing passwords with these algorithms can vulnerability.

**50) Which cryptographic extension provide generation and verification of digital signatures?**

The PHP-openssl extension provides several cryptographic operations including generation and verification of digital signatures.

**51) How a constant is defined in a PHP script?**

The define() directive lets us defining a constant as follows:

define (“ACONSTANT”, 123);

**52) How can you pass a variable by reference?**

To be able to pass a variable by reference, we use an ampersand in front of it, as follows $var1 = &$var2

**53) Will a comparison of an integer 12 and a string “13” work in PHP?**

“13” and 12 can be compared in PHP since it casts everything to the integer type.

**54) How is it possible to cast types in PHP?**

The name of the output type have to be specified in parentheses before the variable which is to be cast as follows:

\* (int), (integer) – cast to integer

\* (bool), (boolean) – cast to boolean

\* (float), (double), (real) – cast to float

\* (string) – cast to string

\* (array) – cast to array

\* (object) – cast to object

**55) When a conditional statement is ended with an endif?**

When the original if was followed by : and then the code block without braces.

**56) How is the ternary conditional operator used in PHP?**

It is composed of three expressions: a condition, and two operands describing what instruction should be performed when the specified condition is true or false as follows:

Expression\_1 ? Expression\_2 : Expression\_3;

**57) What is the function func\_num\_args() used for?**

The function func\_num\_args() is used to give the number of parameters passed into a function.

**58) If the variable $var1 is set to 10 and the $var2 is set to the character var1, what’s the value of $$var2?**

$$var2 contains the value 10.

**59) What does accessing a class via :: means?**

:: is used to access static methods that do not require object initialization.

**60) In PHP, objects are they passed by value or by reference?**

In PHP, objects passed by value.

**61) Are Parent constructors called implicitly inside a class constructor?**

No, a parent constructor have to be called explicitly as follows:

parent::constructor($value)

**62) What’s the difference between \_\_sleep and \_\_wakeup?**

\_\_sleep returns the array of all the variables that need to be saved, while \_\_wakeup retrieves them.

**63) What is faster?**

1- Combining two variables as follows:

$variable1 = ‘Hello ‘;

$variable2 = ‘World’;

$variable3 = $variable1.$variable2;

Or

2- $variable3 = “$variable1$variable2”;

$variable3 will contain “Hello World”. The first code is faster than the second code especially for large large sets of data.

**64) what is the definition of a session?**

A session is a logical object enabling us to preserve temporary data across multiple PHP pages.

**65) How to initiate a session in PHP?**

The use of the function session\_start() lets us activating a session.

**66) How is it possible to propagate a session id?**

It is possible to propagate a session id via cookies or URL parameters.

**67) What is the meaning of a Persistent Cookie?**

A persistent cookie is permanently stored in a cookie file on the browser’s computer. By default, cookies are temporary and are erased if we close the browser.

**68) When sessions ends?**

Sessions automatically ends when the PHP script finishs executing, but can be manually ended using the session\_write\_close().

**69) What is the difference between session\_unregister() and session\_unset()?**

The session\_unregister() function unregister a global variable from the current session and the session\_unset() function free all session variables.

**70) What does $GLOBALS means?**

$GLOBALS is associative array including references to all variables which are currently defined in the global scope of the script.

**71) What does $\_SERVER means?**

$\_SERVER is an array including information created by the web server such as paths, headers, and script locations.

**72) What does $\_FILES means?**

$\_FILES is an associative array composed of items sent to the current script via the HTTP POST method.

**73) What is the difference between $\_FILES[‘userfile’][‘name’] and $\_FILES[‘userfile’][‘tmp\_name’]?**

$\_FILES[‘userfile’][‘name’] represents the original name of the file on the client machine,

$\_FILES[‘userfile’][‘tmp\_name’] represents the temporary filename of the file stored on the server.

**74) How can we get the error when there is a problem to upload a file?**

$\_FILES[‘userfile’][‘error’] contains the error code associated with the uploaded file.

**75) How can we change the maximum size of the files to be uploaded?**

We can change the maximum size of files to be uploaded by changing upload\_max\_filesize in php.ini.

**76) What does $\_ENV means?**

$\_ENV is an associative array of variables sent to the current PHP script via the environment method.

**77) What does $\_COOKIE means?**

$\_COOKIE is an associative array of variables sent to the current PHP script using the HTTP Cookies.

**79) what the difference between the ‘BITWISE AND’ operator and the ‘LOGICAL AND’ operator?**

$a and $b:    TRUE if both $a and $b are TRUE.

$a & $b:        Bits that are set in both $a and $b are set.

**80) What are the two main string operators?**

The first is the concatenation operator (‘.’), which returns the concatenation of its right and left arguments. The second is (‘.=’), which appends the argument on the right to the argument on the left.

**81) What does the array operator ‘===’ means?**

$a === $b TRUE if $a and $b have the same key/value pairs in the same order and of the same types.

**82) What is the differences between $a != $b and $a !== $b?**

!= means inequality (TRUE if $a is not equal to $b) and !== means non-identity (TRUE if $a is not identical to $b).

**83) How can we determine whether a PHP variable is an instantiated object of a certain class?**

To be able to verify whether a PHP variable is an instantiated object of a certain class we use instanceof.

**84) What is the goto statement useful for?**

The goto statement can be placed to enable jumping inside the PHP program. The target is pointed by a label followed by a colon, and the instruction is specified as a goto statement followed by the desired target label.

**85) what is the difference between  Exception::getMessage and Exception::getLine ?**

Exception::getMessage lets us getting the Exception message and Exception::getLine lets us getting the line in which the exception occurred.

**86) What does the expression Exception::\_\_toString means?**

Exception::\_\_toString gives the String representation of the exception.

**87) How is it possible to parse a configuration file?**

The function parse\_ini\_file() enables us to load in the ini file specified in filename, and returns the settings in it in an associative array.

**88) How can we determine whether a variable is set?**

The boolean function isset determines if a variable is set and is not NULL.

**89) What is the difference between the functions strstr() and stristr()?**

The string function strstr(string allString, string occ) returns part of allString from the first occurrence of occ to the end of allString. This function is case-sensitive. stristr() is identical to strstr() except that it is case insensitive.

**90) what is the difference between for and foreach?**

for is expressed as follows:

for (expr1; expr2; expr3)

statement

The first expression is executed once at the beginning. In each iteration, expr2 is evaluated. If it is TRUE, the loop continues and the statements inside for are executed. If it evaluates to FALSE, the execution of the loop ends. expr3 is tested at the end of each iteration.

However, foreach provides an easy way to iterate over arrays and it is only used with arrays and objects.

**91) Is it possible to submit a form with a dedicated button?**

It is possible to use the document.form.submit() function to submit the form. For example: <input type=button value=”SUBMIT” onClick=”document.form.submit()”>

**92) What is the difference between ereg\_replace() and eregi\_replace()?**

The function eregi\_replace() is identical to the function ereg\_replace() except that it ignores case distinction when matching alphabetic characters.

**93) Is it possible to protect special characters in a query string?**

Yes, we use the urlencode() function to be able to protect special characters.

**94) What are the three classes of errors that can occur in PHP?**

The three basic classes of errors are notices (non-critical), warnings (serious errors) and fatal errors (critical errors).

**95) What is the difference between characters \034 and \x34?**

\034 is octal 34 and \x34 is hex 34.

**96) How can we pass the variable through the navigation between the pages?**

It is possible to pass the variables between the PHP pages using sessions, cookies or hidden form fields.

**97) Is it possible to extend the execution time of a php script?**

The use of the set\_time\_limit(int seconds) enables us to extend the execution time of a php script. The default limit is 30 seconds.

**98) Is it possible to destroy a cookie?**

Yes, it is possible by setting the cookie with a past expiration time.

**99) What is the default session time in php?**

The default session time in php is until closing of browser

**100) Is it possible to use COM component in PHP?**

Yes, it’s possible to integrate (Distributed) Component Object Model components ((D)COM) in PHP scripts which is provided as a framework.

**101) Explain whether it is possible to share a single instance of a Memcache between multiple PHP projects?**

Yes, it is possible to share a single instance of Memcache between multiple projects. Memcache is a memory store space, and you can run memcache on one or more servers. You can also configure your client to speak to a particular set of instances. So, you can run two different Memcache processes on the same host and yet they are completely independent. Unless, if you have partitioned your data, then it becomes necessary to know from which instance to get the data from or to put into.

**102) Explain how you can update Memcached when you make changes to PHP?**

When PHP changes you can update Memcached by

• **Clearing the Cache proactively:** Clearing the cache when an insert or update is made  
•  **Resetting the Cache:** It is similar to the first method but rather than just deleting the keys and waiting for the next request for the data to refresh the cache, reset the values after the insert or update.