**PHP hypertext preprocessor**

1. **Does PHP support multiple inheritances?**

PHP does not support multiple inheritances. To implement the features of multiple inheritances, the interface is used in PHP.

PHP supports only single inheritance; it means that a class can be extended from only one single class using the keyword 'extended'.

1. **What are Traits?**

Traits are a mechanism that allows you to create reusable code in PHP where multiple inheritance is not supported. To create a Traits we use keyword trait.

1. **How to start displaying errors in PHP application?**

Add following code in PHP.

ini\_set(‘display\_errors’, 1);  
ini\_set(‘display\_startup\_errors’, 1);  
error\_reporting(E\_ALL);

OR Add following code in .htacess

php\_flag display\_startup\_errors on  
php\_flag display\_errors on  
php\_flag html\_errors on  
php\_flag log\_errors on

1. **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 overridden.

1. **How is the comparison of objects done in PHP?**

We use the operator '==' to test is two objects are instanced from the same class and have same attributes and equal values. We can test if two objects are referring to the same instance of the same class by the use of the identity operator '==='.

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

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

1. **Is it possible to protect special characters in a query string?**

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

1. **What is needed to be able to use image function?**

GD library is needed to execute image functions.

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

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

1. **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())

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

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

1. **What is the function file\_get\_contents() useful for?**

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

1. **Which function you can use in PHP to open a file for reading or writing or for both?**

You can use **fopen()** function to read or write or for doing both in PHP.

**Sample code:**

|  |  |
| --- | --- |
|  | $file1 = fopen("myfile1.txt","r"); //Open for reading |
|  | $file2 = fopen("myfile2.txt","w"); //Open for writing |

|  |  |
| --- | --- |
|  | $file3 = fopen("myfile3.txt","r+"); //Open for reading and writing |

1. **How can we connect to a MySQL database from a PHP script?**

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

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

1. **What are the differences between mysqli\_connect and mysqli\_pconnect?**

**mysqli\_pconnect()**function is used for making a persistence connection with the database that does not terminate when the script ends.

**mysqli\_connect()** function searches any existing persistence connection first and if no persistence connection exists, then it will create a new database connection and terminate the connection at the end of the script.

**Sample code:**

|  |  |  |
| --- | --- | --- |
|  | $DBconnection = mysqli\_connect("localhost","username","password","dbname"); | |
|  | // Check for valid connection |

|  |  |  |
| --- | --- | --- |
|  | if (mysqli\_connect\_errno()) | |
|  | { |

|  |  |  |
| --- | --- | --- |
|  | echo "Unable to connect with MySQL: " . mysqli\_connect\_error(); | |
|  | } |

**mysqli\_pconnect()** function is depreciated in the new version of PHP, but you can create persistence connection using mysqli\_connect with the prefix p.

1. **How be the result set of Mysql handled in PHP?**

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

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

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

1. **Which function gives us the number of affected entries by a query?**

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

1. **What is the difference between mysqli\_fetch\_object() and mysqli\_fetch\_array()?**

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

1. **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.

1. **What does the unlink() function mean?**

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

1. **What does the unset() function mean?**

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

1. **Is it possible to remove the HTML tags from data?**

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

1. **What is the use of strip\_tags() method?**

**strip\_tags()** function is used to retrieve the string from a text by omitting HTML, XML and PHP tags. This function has one mandatory parameter and one optional parameter. The optional parameter is used to accept particular tags.

1. **How can a cross-site scripting attack be prevented by PHP?**

Htmlentities() function of PHP can be used for preventing cross-site scripting attack.

1. **What’s the difference between htmlentities() and htmlspecialchars()?**

htmlspecialchars only takes care of <, >, single quote ‘, double quote ” and ampersand.

htmlentities translates all occurrences of character sequences that have different meaning in HTML.

1. **What is the use of mysqli\_real\_escape\_string() function?**

mysqli\_real\_escape\_string() function is used to escape special characters from the string for using a SQL statement

1. **Which functions are used to remove whitespaces from the string?**

There are three functions in PHP to remove the whitespaces from the string.

**trim()** – It removes whitespaces from the left and right side of the string.  
**ltrim()** – It removes whitespaces from the from the left side of the string.  
**rtrim()** – It removes whitespaces from the from the right side of the string.

1. **How is a constant defined in a PHP script?**

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

define ("ACONSTANT", 123);

1. **How can you declare a constant variable in PHP?**

**define()** function is used to declare a constant variable in PHP. Constant variable declares without the $ symbol.

1. **When is a conditional statement ended with endif?**

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

1. **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;

1. **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.

1. **What does accessing a class via :: means?**

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

1. **Are Parent constructors called implicitly inside a class constructor?**

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

parent::constructor($value

1. **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.

1. **How to initiate a session in PHP?**

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

1. **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.

1. **What is a persistence cookie?**

A cookie file that is stored permanently in the browser is called a persistence cookie. It is not secure and is mainly used for tracking a visitor for long times.

This type of cookie can be declared as follows,

setccookie ("cookie\_name", "cookie\_value", strtotime("+2 years");

1. **Is it possible to destroy a cookie?**

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

1. **When do sessions end?**

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

1. **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 frees all session variables.

1. **What does $GLOBALS mean?**

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

1. **What does $\_SERVER mean?**

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

1. **What does $\_FILES means?**

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

1. **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.

1. **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.

1. **Which PHP global variable is used for uploading a file?**

$\_FILE[] array contains all the information of an uploaded file.

**The use of various indexes of this array is mentioned below:**

**$\_FILES[$fieldName][‘name']** – Keeps the original file name.  
**$\_FILES[$fieldName][‘type']** – Keeps the file type of an uploaded file.  
**$\_FILES[$fieldName][‘size']** – Stores the file size in bytes.  
**$\_FILES[$fieldName][‘tmp\_name']** – Keeps the temporary file name which is used to store the file in the server.  
**$\_FILES[$fieldName][‘error']** – Contains error code related to the error that appears during the upload.

1. **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.

1. **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.

1. **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.

1. **What does $\_ENV mean?**

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

1. **What does $\_COOKIE mean?**

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

1. **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.

1. **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.

1. **What is the difference between for and foreach loop in PHP?**

**for** loop is mainly used for iterating a pre-defined number of times and **foreach** loop is used for reading array elements or MySQL result set where the number of iteration can be unknown.

1. **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()">

1. **What is meant by public, private, protected, static and final scopes?**
2. **Public**– Variables, classes, and methods which are declared public can be accessed from anywhere.
3. **Private**– Variables, classes and methods which are declared private can be accessed by the parent class only.
4. **Protected**– Variables, classes, and methods which are declared protected can be accessed by the parent and child classes only.
5. **Static**– The variable which is declared static can keep the value after losing the scope.
6. **Final**– This scope prevents the child class to declare the same item again.
7. **What is PDO?**

The full form of PDO is PHP Data Objects.

It is a lightweight PHP extension that uses consistence interface for accessing the database. Using PDO, a developer can easily switch from one database server to the other. But it does not support all the advanced features of the new MySQL server.

1. **What’s the output of the ucwords function in this example?**

$formatted = ucwords(“PHP interview questions for experienced”);  
print $formatted;

A) ucwords() makes every first letter of every word capital.

1. **What are the different functions in sorting an array?**

Sorting functions in PHP:

1. asort()
2. arsort()
3. ksort()
4. krsort()
5. uksort()
6. sort()
7. natsort()
8. rsort()
9. **How can we know the count/number of elements of an array?**

In two ways, we can count the number of elements of an array:  
a) sizeof($array) – This function is an alias of count()  
b) count($urarray) – This function returns the number of elements in an array.

1. **How we use ceil() and floor() function in PHP?**

ceil() is use to find nearest maximum values of passing value.

Ceil Example:  
$var=6.5;  
$ans\_var=ceil($var);  
echo $ans\_var;

Output: 7

floor() is use to find nearest minimum values of passing value.

Floor Example:  
$var=6.5  
$ans\_var=floor($var);  
echo $ans\_var;

1. **What is the answer of following code echo 1< 2 and echo 1 >2 ?**

Output of the given code are given below:  
echo 1<2  
output: 1

echo 1>2  
output: no output

1. **The main difference b/w isset and empty are:**

isset: This variable is used to handle functions and checked a variable is set even through it is empty.

empty: This variable is used to handle functions and checked either variable has a value or it is an empty string,zero0 or not set at all.

1. **How to redirect https to HTTP URL and vice versa in .htaccess?**

Redirect https to http

RewriteEngine On  
RewriteCond %{HTTPS} on  
RewriteRule (.\*) http://%{HTTP\_HOST}%{REQUEST\_URI} [R=301,L]

Redirect http to https

RewriteEngine on  
RewriteCond %{HTTPS} off  
RewriteRule ^(.\*)$ https://%{HTTP\_HOST}/$1 [R=301,L]

1. **What is the difference between abstract class and interface?**
2. **What is URL rewriting?**
3. **What is Singleton pattern and MVC pattern?**
4. **What is the SOAP and REST?**
5. **What is WSDL?**
6. **What are the required parameters for Paypal integration?**
7. **How to improve the performance of the application?**
8. **How you provide the security to your application?**
9. **Serialzation?**
10. **Designpatterns?**
11. **MVC?**
12. **What is MVC?**
13. **What is use of MVC?**
14. **Call by Value and Call by Reference in PHP?**
15. **singlesignon?**
16. **how many way we can pass one variable to another page?**
17. **file functions (fread ,fwrite,fsize,permissions)?**
18. **Framework vs Library?**
19. **what is a namespace,clousers and Traits?**
20. **REST and SOAP**
21. **Magic methods**
22. **httpd conf**
23. **PHP.ini**
24. **CURL**
25. **Overloading vs overriding**
26. **abstract class vs interface? where you will use?**
27. **Errors and Exception Handling and examples?**
28. **Final and Static**
29. **Interfaces and Abstract class**

**GIT**

1. **Create GIT Repo & Branch & Track Branch**  
   1. git clone <http://xxxxxxx.git>
   2. git branch -l
   3. git status
   4. git checkout master
   5. git pull
   6. git fetch --all;
   7. git checkout develop
   8. git pull
   9. git checkout <branch\_name>
   10. git branch –l
   11. git pull
   12. git status
   13. git add file-names
   14. git commit - m ‘comments’
   15. git diff --name-only master..<branch\_name>
   16. git merge master
   17. git diff --name-only master..<branch\_name>
   18. git push origin feature\_name
2. **List Of Branch in Repo:** git branch -l
3. **For Config file View:** git config -e
4. **For Cancel The Commit:** git reset --hard xxxxxxxxxx
5. **For Git Push:** *git push origin <feature\_name>*
6. **For List of files:** *git diff --name-only master...<veera\_tickets2>*
7. **Tag creation:**

1). git checkout stage

2). git branch

3). git fetch --all;

4). git pull origin stage;

5). git merge e9c674e9047246c723cc6dc9a4cb28e158ee3563

6). git describe --tags

7). git tag -a WS1.0\_STG\_CIA\_167\_V0 -m "Creating tag for CIA 167"

8). git push origin WS1.0\_STG\_CIA\_167\_V0

9). git push origin stage

10). git diff WS1.0\_STG\_CIA\_167\_V0..WS2.0\_STG\_CIA\_070\_V0 --name-only

11). git tag -d WS1.0\_STG\_CIA\_167\_V0;

12). git push --delete origin WS1.0\_STG\_CIA\_167\_V0

**Command Line**

* 1. **Set Proxy in Command line:**

set http\_proxy=http://proxy.xxxxx.com:6050

set https\_proxy=http://proxy.xxxxx.com:6050

* 1. **Database Export:**

mysql -u root -p webshop < D:\AbbottFiles\dump.sql

**MySQL**

1. **How can we repair a MySQL table?**

We can use REPAIR command to repair a table. The REPAIR command will repair the table specified.

The syntex for repairing a mysql table is:  
REPAIR TABLE tablename  
REPAIR TABLE tablename QUICK  
REPAIR TABLE tablename EXTENDED

If QUICK is given, MySQL will do a repair of only the index tree.  
If EXTENDED is given, it will create index row by row.

1. **What are the different tables present in MySQL?**

A) Total 5 types of tables we can create

1. MyISAM
2. Heap
3. Merge
4. INNO DB
5. ISAM
6. **What is the maximum length of a table name, a database name, or a field name in MySQL?**

The maximum length of a table name, database name and field name is:

Database name: 64 characters  
Table name: 64 characters  
Column name: 64 characters

1. **What are the other commands to know the structure of a table using MySQL commands except EXPLAIN command?**

DESCRIBE table\_name;

1. **What’s the difference between md5(), crc32() and sha1() crypto on PHP?**

The major difference is the length of the hash generated.

CRC32 is, evidently, 32 bits, while sha1() returns a 128 bit value, and md5() returns a 160 bit value.

This is important when avoiding collisions.

1. **How many ways we can we find the current date using MySQL?**

We can find current date using MySQL in different ways, they are:

* SELECT CURDATE();
* SELECT CURRENT\_DATE();
* SELECT CURTIME();
* SELECT CURRENT\_TIME();

1. **How can we know the number of days between two given dates using MySQL?**

Use DATEDIFF()  
SELECT DATEDIFF(NOW(),’2006-07-01′);

1. **How can we change the data type of a column of a table?**

ALTER TABLE table\_name CHANGE colm\_name same\_colm\_name [new data type]

1. **How to get the 2nd highest salary of an employee, if two employees may have the same salary?**

select salary from employee group by salary order by salary limit 1,1

1. **How to find duplicate email records in users table?**

SELECT u1.first\_name, u1.last\_name, u1.email FROM users as u1  
INNER JOIN (  
SELECT email FROM users GROUP BY email HAVING count(id) > 1  
) u2 ON u1.email = u2.email;

1. **Vulnerabilities in MySQL database**

Cross Site Scripting   
Cross Site Request Forgery  
Sql Injection  
Weak Password.  
Database Inconsistency  
Denial of Services.  
Buffer Overflow ( Stack based and Heap Based ) Execution of arbitrary code

**How to recover :-**Disable Remote Access  
Don't run MySQL as root user  
Set a password for root account  
Remove anonymous user accounts  
Don't grant all users PROCESS or SUPER privilege  
Don't grant all users the FILE privilege  
Restrict access to log files  
Remove Content of MySQL History File  
Keeping MySQL Version Up to Date  
Yassl/openssl while configuring

1. **Normalization**
2. **Primary key and Unique key**
3. **Inner join vs outer join**
4. **Group by vs having clause**
5. **Index(primary,unique,index,fulltext)**
6. **Joins(inner,outer)**
7. **Union/Union All**
8. **Types of DB engines**
9. **MyIsam vs Innodb**
10. **how many ways we can retrive data from mysql resultset**
11. **constraints (primary, foreign key, unique, not null,Default)**
12. **Trigger**
13. **Cursor**
14. **Stored Procedure vs function**
15. **Trigger vs stored procedure vs functions**
16. **order by and group by**

**Update with replace string:**

UPDATE core\_config\_data

SET value = REPLACE(value, 'fsl.abbott.in/', 'localhost:8080/adcwebshop/')

WHERE value LIKE '%fsl.abbott.in/%'

**Drupal**

1. **How to create a theme?**
2. **Tell me different types of theme pre-process functions?**
3. **How to create a custom hook?**
4. **Can we have multiple submit handlers for a form in Drupal?**
5. **How to add head meta tags before processing a page?**
6. **How to suggest a layout for a node page?**
7. **What are the differences between d6 and d7?**
8. **What are the different types of hooks?**
9. **How to suggest a menu type for  menu?**
10. **What is view?**
11. **What is feature in theme.info?**
12. **How to create a module in Drupal?**
13. **What is model?**
14. **Do you use any template engine in MVC?**
15. **How you set SMARTY in mvc frame work?**
16. **How you pass the data to smarty?**
17. **Why we not use curl instead of restful for web services?**
18. **How can we provide user authentication for web services and how we pass authentication details?**

AngularJS

1. **What is the Angular version you are using ?**
2. **Scope in angular ?**
3. **What is the difference between $scope and scope.**
4. **Link and compare in Angular ?**
5. **Broadcasting in Angular ?**
6. **Directives in Angular ?**
7. **Different directives in angular ?**
8. **How to create custom directive ?**
9. **What is the difference between Angular expression and JavaScript expression ?**
10. **What is RootScope ?**
11. **What is an isolated Scope ?**
12. **What is a class in Angular ?**
13. **How a class in used in Angular ?**
14. **What is a template in angular ?**
15. **Does behavior mean in Angular ?**
16. **Closure in javascript ?**
17. **What does .call() and .apply() do in javascript ?**
18. **Inheritance in JavaScript ?**

**CSS**

paddding and margin

display vs visibility

place 3 divs horizontal and vertical

mysql:

1)diff btwn delete and truncate

DELETE:deletes table rows and returns number of rows deleted.

We can delete selected no of records from table.

Delete stmt has 2 modifiers- LOW\_PRIORITY and QUICK

When we use DELETE command then the data delted can be retrieved when you ROLLBACK.

QUICK modifier is specified then the table handler does not merge index leaves during delete,this may cause to speed up certain kind of deletes.

LIMIt clause van be used to limit the no of rows to delete.

ORDER BY used-because rows deleted in specified order.

DELETE FROM student WHERE roll\_no<25;

TRUNCATE:

Drops the table and re-creat it(as an empty table).only tbl\_name.frm is valid.

cannot retrive data again from the table.

removes all the rows from the table.

does not return number of deleted rows.

it is not transaction-safe,error occurs when attempting one in the course of an active transaction or active table lock

After truncate this table do not contains any fields but it show in our database by using the following command

TRUNCATE table\_name;

DROP:deletes the data as well as structure.

The difference between DROP and DELETE table is that, after executing delete statement the contents of table are removed but the structure remains same, but in case of DROP statement both the contents and structure are removed.

DROP TABLE table\_name;

2)query to get 2nd highest salary

SELECT MAX(salary) FROM Employee

WHERE

salary NOT IN(SELECT MAX(salary) FROM Employee );

SELECT \* FROM Employee Emp1

WHERE (N-1)=(SELECT COUNT(DISTINCT(Emp2.salary))

FROM Employee Emp2

WHERE Emp2.salary>Emp1.salary)

select salary from test order by salary limit 1, 1;

php

1)header() use

Ans:header() sends raw HTTP header to a client.

It is important to notice that header() must be called before any actual output is sent (In PHP 4 and later, you can use output buffering to solve this problem):

Ex:header('Location: http://www.example.com/');

Syntax:header(string,replace,http\_response\_code);

Ex:header("Expires: Mon, 26 Jul 1997 05:00:00 GMT");

header("Cache-Control: no-cache");

header("Pragma: no-cache");

Ex2:

header("Content-type:application/pdf");--prompted to save a generated PDF file

what is cookie and session.relation between cookie and session?

The Relationship between Sessions and Cookies

Sessions are not cookies, but they can (and do) work together to create the illusion of persistence in an otherwise stateless protocol. Sometimes persistence is referred to as "stickiness", or "sticky connections." That's because what persistence does is ensure that a client connects to the "real" server on which his/her current session is active.

Persistence is the process of ensuring that a user is connected to the same server every time they make a request within the boundaries of a single session. Even though users could be persisted based on their IP address, this is rarely done due to the sharing of IP addresses. Persistence is most often implemented using a cookie containing the server session id because it is the most accurate method of determining where a user's session is currently stored.

Sessions are stored on the server, and are not reliant on cookies being enable in the client's browser. Sessions are where web developers store bits of application relevant data that they may wish to use across requests. Shopping carts are the most ubiquitous example of session data, but there are other uses for it, especially in complex web applications like CRM (customer relationship management) or SFA (sales force automation) applications.

Cookies store bits of data on the client (the browser) and are passed to the server via the HTTP header Cookie.

.htaccess use

An .htaccess file is a way to configure the details of your website without needed to alter the server config files. The period that starts the file name will keep the file hidden within the folder.

We can do 2 things using .htaccess file

a)Speed:.htaccess file affects the pages in its directory and all of the directories under it.

b)Security

Five Common Uses for an .htaccess Page

a)mod\_rewrite.

You can use the space in the .htaccess file to designate and alter how URLs and web pages on your sites are displayed to your users.

b)Authentication: Although using the .htaccess file does not require as many permissions as accessing the apache2.conf file would require, we can still make effective changes to a site.

c)Custom Error Pages:allows you to create custom error pages.To make a page look friendlier and to provide more information to the site visitor than the default server error page offers, you can use the .htaccess file to create custom error pages.

syntax:ErrorDocument 404/new404.html

d)Mime Types:

AddType audio/mp4a-latm.m4a

e)SSI: Server Side Includes are a great time saver on a website.

One of the most common uses of SSI is to update a large number of pages with some specific data, without having to update each page individually (for example, if you want to change a quotation at the bottom of a page).

inheritance:

abstract class use

Not only can you define a template for children, but Abstract Classes offer the added benefit of letting you define functionality that your child classes can utilize later.

You can't provide an implementation for an Interface.

When should you prefer them over interfaces and when not?

Abstract Classes are a good fit if you want to provide implementation details to your children but don't want to allow an instance of your class to be directly instantiated (which allows you to partially define a class).

If you want to simply define a contract for Objects to follow, then use an Interface.

Also when are abstract methods useful?

Abstract methods are useful in the same way that defining methods in an Interface is useful. It's a way for the designer of the Abstract class to say "any child of mine MUST implement this method".

Answer

Abstract classes are classes that contain one or more abstract methods. An abstract method is a method that is declared, but contains no implementation. Abstract classes may not be instantiated, and require subclasses to provide implementations for the abstract methods. Let's look at an example of an abstract class, and an abstract method.

Suppose we were modeling the behavior of animals, by creating a class hierachy that started with a base class called Animal. Animals are capable of doing different things like flying, digging and walking, but there are some common operations as well like eating and sleeping. Some common operations are performed by all animals, but in a different way as well. When an operation is performed in a different way, it is a good candidate for an abstract method (forcing subclasses to provide a custom implementation). Let's look at a very primitive Animal base class, which defines an abstract method for making a sound (such as a dog barking, a cow mooing, or a pig oinking).

public abstract Animal

{

public void eat(Food food)

{

// do something with food....

}

public void sleep(int hours)

{

try

{

// 1000 milliseconds \* 60 seconds \* 60 minutes \* hours

Thread.sleep ( 1000 \* 60 \* 60 \* hours);

}

catch (InterruptedException ie) { /\* ignore \*/ }

}

public abstract void makeNoise();

}

Note that the abstract keyword is used to denote both an abstract method, and an abstract class. Now, any animal that wants to be instantiated (like a dog or cow) must implement the makeNoise method - otherwise it is impossible to create an instance of that class. Let's look at a Dog and Cow subclass that extends the Animal class.

public Dog extends Animal

{

public void makeNoise() { System.out.println ("Bark! Bark!"); }

}

public Cow extends Animal

{

public void makeNoise() { System.out.println ("Moo! Moo!"); }

}

Now you may be wondering why not declare an abstract class as an interface, and have the Dog and Cow implement the interface. Sure you could - but you'd also need to implement the eat and sleep methods. By using abstract classes, you can inherit the implementation of other (non-abstract) methods. You can't do that with interfaces - an interface cannot provide any method implementations.

Example of Abstract:

<?php

abstract class Vehicle {

public abstract function getNumWheels();

public function getName() {

return get\_class($this);

}

}

class Car extends Vehicle {

public function getNumWheels() {

return 4;

}

}

class Bike extends Vehicle {

public function getNumWheels() {

return 2;

}

}

function printNumWheels(Vehicle $v) {

echo "A " . $v->getName() . " has " . $v->getNumWheels() . " wheels\n";

}

$car = new Car();

$bike = new Bike();

printNumWheels($car);

printNumWheels($bike);

/\*

output:

A Car has 4 wheels

A Bike has 2 wheels

\*/

interfaces

diffrent types of inheritance

php.ini execution time max upload size of image

increse execution time:ini\_set('max\_execution\_time',300)

maximum upload file size:

ini\_set('post\_max\_size','64M');

ini\_set('upload\_max\_filesize','64M');

file systems how move a file

http methods(\_get,\_post,\_request)

.httpd.conf file

regular expressions

jquery

how to call an attribute by name,id,class

syntax of jquery

$('input').(first);

Interview Questions

1) Agile

a. What is Agile Methodology?

b. What is Sprint?

c. What is Scrum in Agile?

d. What is Poker Estimation?

e. How do you create Sub Tasks in JIRA?

f. Explain how you implemented agile methodology in your project?

2) MySQL

a. What are Primary Key, Candidate Key and Unique Key? Differences between them?

b. What is the default MySQL engine?

c. What is the difference between InnoDB and MYISAM database engines

3) JQuery

a. If I have 5 $(document).ready(function() in which order the functions are called ?

b. What is Ajax and how you call Ajax in JQuery – need syntax?

c. What is “async” attribute in JQuery?

4) OOP’s

a. What is inheritance?

b. Is Multiple Inheritance supported in PHP? How is this achieved?

c. How do you call Base Class function in Super Class by creating object of Super Class?

5) PHP

a. What is array\_walk()? What is array\_map? What is the difference between them?

b. What is magic functions in PHP?

c. What is \_\_autoload() in PHP?

6) Drupal

a. What is MEM Cache in Drupal?

b. How do you create modules in Drupal?

c. What is .info, .module and .install files in Module?

d. What are hooks in Drupal and tell some hooks in Drupal?