

Club Code Training

វគ្គសិក្សាបង្កើត គេហទំព័រ

PHP Web Development

Introduction

PHP ត្រូវបានផ្តល់ឈ្មោះជាផ្លូវការ Hypertext Preprocessor ជាភាសាដែលតំណើរការនៅលើ Server ដែលជា ទូទៅ ត្រូវបាន សរសេរក្នុងបរិបទ HTML ប៉ុន្តែមិនដូចជា HTML page ធម្មតានោះទេ PHP script មិនត្រូវបានបញ្ជូន ទៅអោយ client ដោយ server ផ្ទាល់នោះទេ វាធ្វើការបញ្ជូនដោយ PHP engine ។ PHP code ដែលស្ថិតនៅក្នុង script អាចប្រើដើម្បីធ្វើ ការជាមួយ Databases, បង្កើត ជាប្រភព អាន និងបញ្ចូលទិន្នន័យ ឬ កែប្រែអត្ថបទដែលមាននៅខាងក្នុង files ធ្វើការ ទំនាក់ទំនងទៅ កាន់ Remote Server ព្រមជាមួយនឹងលទ្ធភាពជាច្រើនផ្សេងទៀត ។ PHP is an open-source, interpreted, and object-oriented scripting language that can be executed at the server-side . PHP was created by **Rasmus Lerdorf in 1994** .

- **PHP echo**

PHP echo ប្រើសម្រាប់ output ទិន្នន័យមកលើ browser

```
<?php  
echo "Hello by PHP echo";  
?>
```

Output:

```
Hello by PHP echo
```

• PHP variable

Variable គឺជាគ្រឹះដ៏សំខាន់សំរាប់ភាសាកុំព្យូទ័រវាត្រូវបានគេប្រើដើម្បីផ្ទុកនូវតំលៃជាបណ្តាសន្ននៅពេលដែលប្រតិបត្តិការ Script ម្តងៗ។

សម្រាប់ភាសា PHP ឈ្មោះរបស់ variable ត្រូវតែចាប់ផ្តើមដោយនិមិត្តសញ្ញា\$ (dollar sign) អ្នកអាចផ្តល់នូវឈ្មោះរបស់ variable ជាតួអក្សរ តំលៃលេខ ឬ underscore (_) ក៏ប៉ុន្តែអ្នកមិនអាចប្រើអក្សរដកឃ្លាជាមួយឈ្មោះរបស់ variable នោះទេ ។

Syntax:

```
$variablename=value;
```

PHP Variable: Declaring string, integer, and float

```
<?php
$str="hello string";
$x=200;
$y=44.6;
echo "string is: $str <br/>";
echo "integer is: $x <br/>";
echo "float is: $y <br/>";
?>
```

Output:

```
string is: hello string
integer is: 200
float is: 44.6
```

• PHP Constant Variable

PHP constants គឺជាឈ្មោះ ឬអត្តសញ្ញាណដែលមិនអាចផ្លាស់ប្តូរបានក្នុងអំឡុងពេលដំណើរការ Script ។

PHP constants can be defined by 2 ways:

1. Using **define()** function

Syntax: **define(name, value, case-insensitive)**

2. Using **const** keyword

```
<?php
define("MESSAGE","Hello JavaTpoint PHP");
echo MESSAGE;
?>
```

Output:

```
Hello JavaTpoint PHP
```

- **PHP Control statement**

1. IF Statement

2. IF else Statement

3. IF else IF Statement

4. Switch Statement

• PHP Loop Statement

Loop statement អាចអោយអ្នកអនុវត្តន៍នូវការងារម្តងហើយម្តងទៀតនៅក្នុង program របស់អ្នក រហូតដល់វាសំរេចដល់លក្ខខណ្ឌ ឬអ្នកបញ្ជាអោយចេញពី LOOP ។

1. For Loop
2. Foreach Loop
3. while loop
4. Do while loop

* * Break , Continue

- # For Loop

Syntax:

For(startvalue; condition; incre/decre)

{

 //code

}

Example

```
<?php
for($n=1;$n<=10;$n++){
    echo "$n<br/>";
}
?>
```

- **Foreach Loop**

Syntax1:

```
foreach($array as $value)
{
    //code
}
```

Syntax2:

```
foreach($array as $key=>$value)
{
    //code
}
```

- **while Loop**

Syntax:

```
While(condition)
{
    //code
    incre/decre;
}
```

PHP While Loop Example

```
<?php
$n=1;
while($n<=10){
    echo "$n<br/>";
    $n++;
}
?>
```

- **Do While Loop**

Syntax:

```
do
{
    //code
    incre/decre;

} While(condition)
```

```
<?php
    $x = 1;
    do {
        echo "1 is not greater than 10.";
        echo "</br>";
        $x++;
    } while ($x > 10);
    echo $x;
?>
```

Output:

```
1 is not greater than 10.
2
```

• PHP Array

Array គឺជា variable ពិសេសដែលអនុញ្ញាតអោយអ្នក ផ្ទុកនូវតំលៃបានច្រើននៅក្នុង variable មួយ ។ រាល់តំលៃនីមួយៗត្រូវបានរក្សាទុកនៅក្នុង index របស់ array ដែលអាច ជាលេខ ឬ ជាតួអក្សរ ។ ជា default ធាតុរបស់ array ដែលជា index គឺចាប់ផ្តើមពីលេខ 0 ។

There are 3 type of array in PHP:

1. Indexed arrays - Arrays with a numeric index
2. Associative arrays - Arrays with named keys
3. Multidimensional arrays - Arrays containing one or more arrays

- Indexed arrays

Syntax:

```
$array=array("value1","value2","value3");
```

Ex:

```
$color=array("red","green","yellow")
```

- Associative arrays

Syntax:

```
$array=array("key1"=>"value1","key2"=>"value2", "key3"=>"value3");
```

Ex:

```
$student=array("name"=>"Dara", "Gender"=>"Male", "dob"=>"2009")
```

- **Multidimensional Array**

Syntax:

```
$array=array(  
    array("value1","value2","value3"), //index0  
    array("value1","value2","value3"), //index1  
    ..... , //index n  
);
```

Example

```
$Staff = array  
(  
    array(1,"Makara",20000),  
    array(2,"Sokha",500000),  
    array(3,"Veasna",300000)  
);
```

- # PHP Functions

Function គឺជាបណ្តុំនៃ block code ដែលអ្នកបង្កើត ហើយអនុញ្ញាតិអោយអ្នកហៅ យកមកប្រើ នៅក្នុង script របស់អ្នក ។

Syntax:

```
function function_name( para1,para2 )  
{  
    //logical code  
}
```

//call function

=> function_name(arg1,arg2);

- **PHP Include and Require**

Both include and require are identical to each other, except failure.

include បង្កើតតែការព្រមាន ពេលគឺ E_WARNING ហើយបន្តការប្រតិបត្តិនៃស្រ្តីប។

require បង្កើតError ពេលគឺ E_COMPILE_ERROR និងបញ្ឈប់ការប្រតិបត្តិនៃស្រ្តីប។

• PHP Array Function

PHP provides various array functions to access and manipulate the elements of array. The important PHP array functions are given below.

1. `array_change_key_case()`
2. `count()`
3. `sort()`
4. `array_reverse()`
5. `array_search()`

- ## PHP String Function

PHP provides various string functions to access and manipulate strings.

1. `strlen()`
2. `strtolower()`
3. `strtoupper()`
4. `trim()`
5. `md5()`
6. `join()`

- ## PHP Math Function

1. `sqrt()`
2. `pow()`
3. `abs()`

- **PHP Form Handling(GET/POST)**

- **GET request** គឺ Default form request. ទិន្នន័យដែលបានឆ្លងកាត់ការស្នើសុំទទួលបានគឺអាចមើលឃើញនៅលើកម្មវិធីរុករកតាមអ៊ីនធឺណិត (URL) ដូច្នេះវាមិនមានសុវត្ថិភាពទេ។ អ្នកអាចធ្វើទិន្នន័យចំនួនកំណត់តាមរយៈការស្នើសុំ (Get request) ។
- **Post request** ត្រូវបានប្រើប្រាស់យ៉ាងទូលំទូលាយដើម្បីបញ្ជូនទម្រង់ដែលមានទិន្នន័យច្រើនដូចជា file upload, image upload, login form, registration form etc. ។ ទិន្នន័យដែលបានឆ្លងកាត់ Post Request មិនអាចមើលឃើញនៅលើកម្មវិធីរុករកតាមអ៊ីនធឺណិត (URL) ដូច្នេះវាត្រូវបានធានាសុវត្ថិភាព (secured) ។ អ្នកអាចធ្វើទិន្នន័យយ៉ាងច្រើនតាមរយៈការប្រើ Post Request ។

MySQL Database

What is MySQL ?

MySQL ដែលជាប្រព័ន្ធគ្រប់គ្រងមូលដ្ឋានទិន្នន័យ Open Source SQL ដែលពេញនិយមបំផុតត្រូវបានបង្កើតឡើង ចែកចាយ និងគាំទ្រដោយ Oracle ។



SELECT - extracts data from a database

UPDATE - updates data in a database

DELETE - deletes data from a database

INSERT INTO - inserts new data into a database

CREATE DATABASE - creates a new database

ALTER DATABASE - modifies a database

CREATE TABLE - creates a new table

ALTER TABLE - modifies a table

DROP TABLE - deletes a table

- **PRIMARY KEY and FOREIGN KEY**

PRIMARY KEY គឺជួរឈរ ឬជួរឈរដែលមានតម្លៃដែលកំណត់អត្តសញ្ញាណដាច់ដោយឡែកពីជួរនីមួយៗក្នុងតារាង។ តារាងមូលដ្ឋានទិន្នន័យត្រូវតែមានសោចម្បងសម្រាប់ Optim ដើម្បីបញ្ចូល ធ្វើបច្ចុប្បន្នភាព ស្តារ ឬលុបទិន្នន័យចេញពីតារាងមូលដ្ឋានទិន្នន័យ។ Primary Key ប្រើសម្រាប់កំណត់មិនអោយទិន្នន័យកើតឡើងស្ទួនគ្នា។

FOREIGN KEY គឺជាFieldមួយ (ឬបណ្តុំនៃField) ក្នុងតារាងមួយ ដែលវាRefer to Primary keyនៅក្នុងតារាងផ្សេងទៀត ។ តារាងដែលមានForeign Keyត្រូវបានគេហៅថាតារាងកូនហើយតារាងដែលមានPrimary Keyត្រូវបានគេហៅថាតារាងយោង ឬមេ។ Foreign Key អាចប្រើស្ទួនគ្នាបាន ក្នុងការកំណត់ ទិន្នន័យRecord។

SELECT,INSERT,UPDATE,DELETE STATEMENT

-Select Syntax:

```
SELECT column1, column2, ...  
FROM table_name;
```

-Insert Syntax:

```
INSERT INTO table_name (column1, column2, column3,...)  
VALUES (value1, value2, value3,...)
```

-Update Syntax:

```
UPDATE table_name  
SET column1 = value1, column2 = value2, ...  
WHERE condition;
```

-Delete Syntax:

```
DELETE FROM table_name WHERE condition;
```

- **MySQL WHERE Clause**
Syntax

```
SELECT column1, column2, ...  
FROM table_name  
WHERE condition;
```

- **The MySQL AND, OR and NOT Operators**

AND Syntax:

```
SELECT column1, column2, ...  
FROM table_name  
WHERE condition1 AND condition2 AND condition3 ...;
```

OR Syntax:

```
SELECT column1, column2, ...  
FROM table_name  
WHERE condition1 OR condition2 OR condition3 ...;
```

NOT Syntax:

```
SELECT column1, column2, ...  
FROM table_name  
WHERE NOT condition;
```

- **MySQL COUNT(), AVG() and SUM() ,Max(),MIN() Functions**

The **COUNT()** function returns the number of rows that matches a specified criterion.

The **AVG()** function returns the average value of a numeric column.

The **SUM()** function returns the total sum of a numeric column.

The **MIN()** function returns the smallest value of the selected column.

The **MAX()** function returns the largest value of the selected column.

Syntax:

```
SELECT COUNT(column_name)
```

```
FROM table name
```

```
WHERE condition;
```

- **MySQL ORDER BY Keyword**

The ORDER BY keyword ត្រូវបានប្រើដើម្បីតម្រៀបលទ្ធផលក្នុងលំដាប់ឡើង ឬចុះ។

The ORDER BY keyword តម្រៀបកំណត់ត្រាតាមលំដាប់ឡើងតាមលំនាំដើម។ ដើម្បីតម្រៀបកំណត់ត្រាតាមលំដាប់ចុះ ប្រើKeyword DESC ។

Syntax:

```
SELECT column1, column2, ...  
FROM table_name  
ORDER BY column1, column2,  
... ASC | DESC;
```

• MySQL Joins

A **JOIN** clause ត្រូវបានប្រើដើម្បីផ្សំជួរដេកពីតារាងពីរ ឬច្រើន ដោយផ្អែកលើជួរឈរដែលទាក់ទងគ្នារវាងពួកវា។

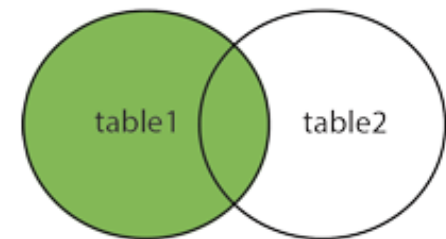
- **LEFT JOIN**
- **INNER JOIN**
- **RIGHT JOIN**

1. **LEFT JOIN** ប្រើដើម្បីត្រឡប់Recordទាំងអស់ពីតារាងខាងឆ្វេង (តារាងទី 1) និងRecordដែលត្រូវគ្នា (ប្រសិនបើមាន) ពីតារាងខាងស្តាំ (តារាងទី 2) ។

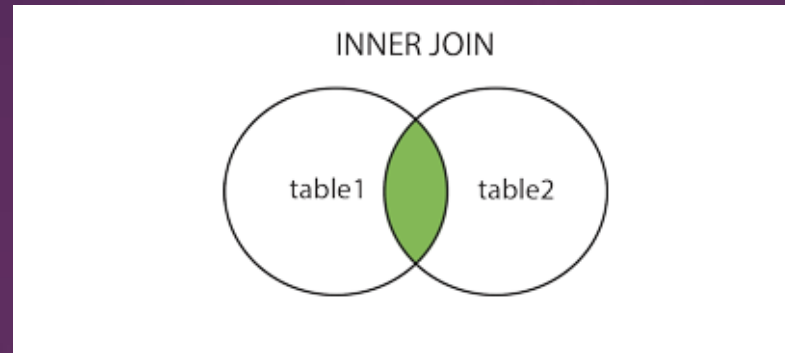
Syntax:

```
SELECT column_name(s)
FROM table1
LEFT JOIN table2
ON table1.column_name = table2.column_name;
```

LEFT JOIN



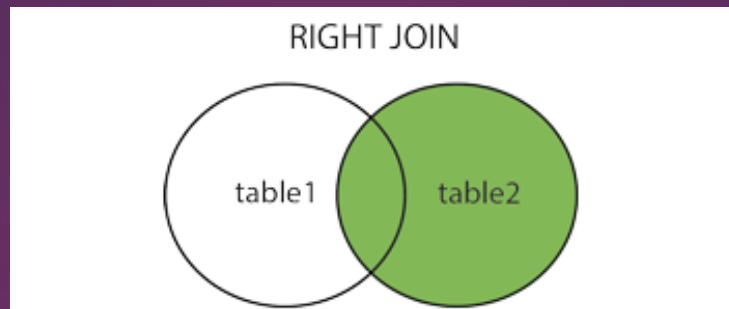
2. INNER JOIN ប្រើសម្រាប់ជ្រើសរើសRecordដែលមានតម្លៃដែលត្រូវគ្នាក្នុងតារាងទាំងពីរ។



Syntax:

```
SELECT column_name(s)  
FROM table1  
INNER JOIN table2  
ON table1.column_name = table2.column_name;
```

3. **RIGHT JOIN** ប្រើដើម្បីត្រឡប់Recordទាំងអស់ពីតារាងខាងស្តាំ (តារាងទី 2) និងRecordដែលត្រូវគ្នា (ប្រសិនបើមាន) ពីតារាងខាងឆ្វេង (តារាងទី 1) ។



Syntax:

```
SELECT column_name(s)  
FROM table1  
RIGHT JOIN table2  
ON table1.column_name = table2.column_name;
```



Thank YOU