1. **In vscode u need to**

**In php file you can write the (HTML,CSS,Javascript,php ) but you can’t write php among of these file. The most powerful is PHP>HTML>CSS**

**Php** is a server site scripting language –bcs if u click the **view page source/inspect element** u can only find the output not **the code it’s so secured** whereas for html & js you can see the code with output.

**পিএইচপির আগে সার্ভার সাইড স্ক্রিপ্টিং সম্পর্কে একটু জানুন:**

আসলে স্ক্রিপ্টিং হচ্ছে প্রোগ্রামের আরেকটা সমার্থক শব্দ। এটা হচ্ছে কিছু instruction এর সেট যেটা run করলে স্বয়ংক্রিয় ভাবে কিছু কাজ হয়। **সার্ভার সাইড** বলতে বুঝানো হচ্ছে এই স্ক্রিপ্ট গুলোকে ইউজারের কম্পিউটার থেকে নিয়ন্ত্রনের বদলে সার্ভার থেকে handle করা। যখন কেউ পিএইচপি ওয়েব পেজ ভিজিট করবে তখন ওয়েব সার্ভার পিএইচপি কোডগুলিকে কিছু Process করবে যেমন:যেটা দেখানো দরকার (Picture,Content etc) সেটা দেখাবে আর যেটাকে লুকানো দরকার (math calculation,file operation etc) তা লুকাবে এবং শেষে HTML এ রুপান্তর করে ইউজারের ওয়েব ব্রাউজারে পাঠাবে।

**পিএইচপি কি?:**

পিএইচপি (PHP:Hypertext Preprocessor ) একটা সার্ভার সাইড স্ক্রিপ্টিং ল্যাংগুয়েজ যেটা মূলত ওয়েব ডেভেলপেমেন্টের জন্য ব্যবহৃত হয়। এই প্রোগ্রামিং ল্যাংগুয়েজটির উদ্দেশ্য হল ওয়েব পেজ দ্রুত তৈরী করা ডাইনামিকালি।

**Variable:**

ভেরিয়েবল হচ্ছে একটা পাত্রের মত(Container) যেখানে আমরা অনেক তথ্য রাখতে পারি। যেমন একটা টেক্সক্ট String “Hello Bangladesh” অথবা একটা integer value 100. কোন একটা ভেরিয়েবল এ একবার তথ্য রেখে সেটা পুরো কোডজুরে বারবার ব্যাবহার করতে পারেন, মুল তথ্য(value) টি বারবার রাখার পরিবর্তে। পিএইচপি তে ভেরিয়েবল “$” এই চিহ্নটি দিয়ে অবশ্যই শুরু করতে হবে নাহলে কাজ করবেনা।একটা ভেরিয়েবলের মান স্ক্রিপ্টে যেকোন সময় পরিবর্তন হতে পারে।

\*ভেরিয়েবল নাম case sensitive.যেমন **$a\_number** and **$A\_number দুটি** **আলাদা** ভেরিয়েবল, পিএইচপি এর দৃষ্টিতে।

\*নিম্নোক্ত ভাবে পিএইচপি তে ভেরিয়েবল লেখা হয়

$variable\_name = Value;

Rules of declaring a Variable:

1. A variable stars with the $ sign, followed by the name of the variable
2. A variable name must start with a letter or the underscore character
3. A variable name cannot start with a number
4. নামের মধ্যে alpha-numeric characters ও underscores. a-z, A-Z, 0-9, or \_ . থাকতে পারে।
5. অবশ্যই কোন letter or “\_”(under score) দিয়ে শুরু করতে হবে।

Data Types:

পিএইচপিতে কয়েক ধরনের ডেটা টাইপ বা ধরন আছে। অর্থ্যাৎ পিএইচপি ভেরিয়েবল এ ধরনের বা টাইপের ডেটা নিতে পারে এবং যেকোন অপারেশন করতে পারে। মুলত পিএইচপির ডেটাকে ২ টি বিভাগে ভাগ করা হয়েছে

**১. স্কেলার ডেটা টাইপ** : এখানে ৪ ধরনের ডেটা টাইপ আছে

ক. boolean বা বুলিয়ান: এ ধরনের ডেটার শুধু ২টি মান থাকতে পারে TRUE অথবা FALSE, পিএইচপিতে অনেক ফাংশন আছে যেগুলি বুলিয়ান রিটার্ন করে। বুলিয়ান ডেটা সাধারনত if কন্ডিশনে ব্যবহার করা হয়। কোন একটা ভেরিয়েবলে বুলিয়ান ডেটা আছে কিনা সেটা দেখার জন্য is\_bool() একটি ফাংশন আছে।

খ. integer বা পূর্নসংখ্যা: পূর্নসংখ্যা তথা দশমিক ছাড়া সংখ্যা হচ্ছে এই ধরনের ডেটা যেমন ৯, ৮, -৬৫, -১ ইত্যাদি হচ্ছে integer ডেটা। is\_int() ফাংশন দিয়ে যেকোন ভেরিয়েবলকে চেক করে দেখতে পারেন, যদি পূর্নসংখ্যা (integer) হয় তাহলে TRUE রিটার্ন করবে অন্যথায় FALSE. পূর্নসংখ্যা ধনাত্নক কিংবা ঋনাত্নক হতে পারে এমনকি দশভিত্তিক (decimal- 10 based) ছাড়াও হেক্সাডেসিমাল (১৬-ভিত্তিক), octal (৮-ভিত্তিক), দ্বিমিক (২-ভিত্তিক বা বাইনারি) সংখ্যাও integer হতে পারে।

গ. Float বা দশমিক সংখ্যা: দশমিক সংখ্যা যেমন ৫.২৩৬, ৫.২ ইত্যাদি হচ্ছে float number বা doubles বা real number

ঘ. String বা অক্ষরের সারি: যেকোন অক্ষর সিঙ্গেল কোট, ডাবল কোটেশন, heredoc এবং nowdoc সিনট্যাক্সের মধ্যে রাখলে সেটা স্ট্রিং যেমন

**২. Compound বা মিশ্র ডেটা টাইপ**

ক. অ্যারে: [পিএইচপি অ্যারে টিউটোরিয়াল (PHP Array )](http://www.webcoachbd.com/php-basic/php-array)

খ. অবজেক্ট: [অবজেক্ট এবং ক্লাস (Object and Class)](http://www.webcoachbd.com/php-framework/object-and-class)

এছাড়া আরো দুটি বিশেষ ডেটা টাইপ আছে resource এবং NULL

resource সাধারনত বিভিন্ন ধরনের ডেটা হয়ে থাকে যেমন ডেটাবেসে অপারেশন করে যেসব ডেটা রিটার্ন করে সেগুলি resource ডেটা হিসেবে দেখে। আর NULL হল ফাকা ডেটা। কোন ভেরিয়েবল ডিক্লেয়ার করা হল কিন্তু কোন মান assign করা হলনা তাহলে সেই ভেরিয়েবলটির মান null হয়ে যায়।

**Operator:**

অপারেটর: যেমন $x + $y; এখানে $x, $y এগুলি হচ্ছে অপারেন্ড (Operand) আর প্লাস (+) চিহ্ন হচ্ছে অপারেটর। আরও অনেক অপারেটর আছে যেমন +, -, \*, /, !, ++, --, ||, and ইত্যাদি।

অনেক অপারেটর আছে যারা শুধু একটি অপারেন্ড এর উপর কাজ করে যেমন ++ (increment অপরেটর) বা ! (not অপারেটর) ইত্যাদি, এসব অপারেটরকে ইউনারি (Unary) অপারেটর বলে।

অনেক অপারেটর আছে যারা দুটি অপারেন্ড এর উপর কাজ করে যেমন + (Addition বা plus অপরেটর) বা - (Subtraction বা minus অপারেটর) ইত্যাদি, এসব অপারেটরকে বাইনারি (Binary) অপারেটর বলে। বেশির ভাগ অপারেটর বাইনারি অপারেটর।

অনেক অপারেটর আছে যারা তিনটি অপারেন্ড এর উপর কাজ করে যেমন ? :, এসব অপারেটরকে টেনারি (Tenary) অপারেটর বলে। টেনারি অপারেটর একটাই।

**অপারেটরের কিছু শ্রেনীবিভাগ আছে যেমন:**

**Arithmetic বা গানিতিক অপারেটর**

ছোটবেলায় অংক করছিলেন যে মনে আছে? ঐসব অংকে যোগ, বিয়োগ, গুন, ভাগ এর জন্য +, -, x চিহ্নগুলি যেভাবে ব্যবহার করেছিলেন ঠিক তেমনি।

উদাহরন অপারেটরের নাম ব্যাখ্যা

-$x Negation $x এর বিপরীত

$x + $y Addition বা যোগ $x এবং $y এর যোগ বরতে ব্যবহৃত হয়েছে

$x - $y Subtraction বা বিয়োগ $x থেকে $y বিয়োগ করতে ব্যবহার হয়েছে

$x \* $y Multiplication বা গুন $x এবং $y এর গুনন করা হয়েছে

$x / $y Division বা ভাগ $x কে $y দিয়ে ভাগ করা হয়েছে

$x % $y Modulus (মডুলাস) $x কে $y দিয়ে ভাগ করার পর অবশিষ্টাংশ

$x \*\* $y Exponentiation বা সূচকীয় পিএইচপি ৫.৬ ভার্সনে এসেছে এখনো ব্যবহার ব্যাপকভাবে শুরু হয়নি।

**Assignment বা আরোপন অপারেটর:** **+=,-=,/=,%=**

এসাইনমেন্ট অপারেটর মুলত সমান (=) চিহ্ন। এর সাথে যুক্ত হয়ে আরো কিছু এসাইনমেন্ট অপারেটর আছে সেগুলিকে বলে Combined Operator বা সংযুক্ত অপারেটর।

**Logical বা যৌক্তিক**অপারেট: || or, && and, ! not

**লজিকাল/Logical বা যৌক্তিক**অপারেটর খুবই গুরত্বপূর্ন যেকোন প্রোগ্রামিং এর ক্ষেত্রে। একের অধিক এক্সপ্রেশন বা শর্ত যাচাই করতে এই অপারেটর ব্যবহার হয়। নিচে তালিকা দেয়া হল

**Comparison Operators:**

The PHP comparison operators are used to compare two values (number or string):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| == | Equal | $x == $y | Returns true if $x is equal to $y |  |
| === | Identical | $x === $y | Returns true if $x is equal to $y, and they are of the same type |  |
| != | Not equal | $x != $y | Returns true if $x is not equal to $y |  |
| <> | Not equal | $x <> $y | Returns true if $x is not equal to $y |  |
| !== | Not identical | $x !== $y | Returns true if $x is not equal to $y, or they are not of the same type |  |
| > | Greater than | $x > $y | Returns true if $x is greater than $y |  |
| < | Less than | $x < $y | Returns true if $x is less than $y |  |
| >= | Greater than or equal to | $x >= $y | Returns true if $x is greater than or equal to $y |  |
| <= | Less than or equal to | $x <= $y | Returns true if $x is less than or equal to $y |  |

PHP Array: array means class or in bangla srani

একটা ভেরিয়েবল একটা single value এর পাত্র(container).কিন্তু অ্যারে একাধিক value’র container বা পাত্র।একটা অ্যারে হল কিছু উপাদান নিয়ে গঠিত যেখানে প্রত্যেকটি উপাদানের নির্দিষ্ট value আছে-key বা index বলে যেটা দিয়ে উপাদানটিকে

রেফার করা হয়।

অ্যারের

সাধারন গঠন পদ্ধতি হচ্ছে কিছু উপাদানের ক্রম (series of element) যার উপাদানগুলির index ০ থেকে শুরু হয়ে ক্রমানুসারে বাড়তে থাকবে।বেশ কয়েকভাবে অ্যারে লেখা যায় সবচেয়ে সহজটি হচ্ছে-

1.<?php

2.

3.$city = array("Dhaka", "Chittagong", "Rajshahi","Sylet", "Khulna", "Barishal");

4.

5.?>

ব্যাস একটা variable এ সব শহরের নাম store হয়ে গেছে,প্রতিটি শহরের নাম এক একটি উপাদান এবং প্রতিটি উপাদানে একটি index নির্দিষ্ট হয়ে গিয়েছে। array’র ১ম উপাদান এর index/key হচ্ছে ০ তাহলে এখানে Dhaka[0], Chittagong[1]… এভাবে বাকিগুলো।এখন আপনি যেকোন উপাদান এর index number উল্লেখ করে সে উপাদানটি পেতে পারেন।যেমন

1.<?php

2.

3.Print "$city[3]";

4.

5.?>

তবে যদি আমরা চাই তাহলে অ্যারের এই ডিফল্ট ভ্যালু পরিবর্তন করে দিতে পারি,=> এই অপারেটরটি দিয়ে।ধরি আমি চাচ্ছি এই উপরের অ্যারেটির starting index 2 দিয়ে শুরু হোক তাহলে এভাবে লিখতে হবে।

1.<?php

2.

3.$city=array(2=>"Dhaka", "Chittagong", "Rajshahi",

4.

5."Sylet", "Khulna", "Barishal", "Barishal");

6.

7.?>

এই অপারেটর যেকোন উপাদানের পূর্বে ব্যাবহৃত হতে পারে তার index নির্দিষ্ট করার জন্য।

In PHP, there are three types of arrays:

1. **Indexed arrays** - Arrays with a numeric index:There are two ways to create indexed arrays:

The index can be assigned automatically (index always starts at 0), like this:

$cars = array("Volvo", "BMW", "Toyota");

or the index can be assigned manually:

$cars[0] = "Volvo";

$cars[1] = "BMW";

$cars[2] = "Toyota";

1. **Associative arrays** - Arrays with named keys

PHP Associative Arrays

Associative arrays are arrays that use named keys that you assign to them.

There are two ways to create an associative array:

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

or:

$age['Peter'] = "35";

$age['Ben'] = "37";

$age['Joe'] = "43";

The named keys can then be used in a script:

Example

<?php

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

echo "Peter is " . $age['Peter'] . " years old.";

?>

1. **Multidimensional arrays** - Arrays containing one or more arrays

এখানে একটা অ্যারের ভিতর প্রতিটি উপাদান একটি অ্যারে হতে পারে আবার এই সাব অ্যারের প্রতিটি উপাদানও একটি অ্যারে হতে পারে এবং এভাবে আরও।

<?php

$cars =array(

'volvo'=>array('price'=>30,'color'=>'Red','Brand'=>'Foreign'),

'BMW'=>array('price'=>40,'color'=>'Black','Brand'=>'Indian'),

'Toyota'=>array('price'=>50,'color'=>'White','Brand'=>'USA'),

);

echo $cars['volvo'] ['color'];

?>

Result : RED

**PHP loop:**

**Loop means repeat(again & again)**

সব ল্যাংগুয়েজের মত পিএইচপিতেও লুপ একই জিনিস। লুপ দিয়ে একটা কোডব্লক যতবার ইচ্ছা এক্সিকিউট করানো যায়। পূনরাবৃত্তির কাজগুলি লুপ দিয়ে করা হয়। লুপে শর্ত দেয়া যায় এবং যতক্ষন শর্ত না মিলবে ততক্ষন পূনরাবৃত্তি চলতেই থাকবে। যেমন ধরুন ১ থেকে ১০০ পর্যন্ত দেখতে চান তাহলে সেটা করতে পারেন মাত্র কয়েক লাইনেই, লুপ দিয়ে। ওয়েব এপ্লিকেশন তৈরীর সময় এরুপ হাজার হাজার অবস্থা আসবে যেখানে আপনাকে লুপ ব্যবহার করতে হবে।

যেমন

- একটা ড্রপডাউন তৈরী করতে হবে যেখানে ১-১২ পর্যন্ত থাকবে যাতে ইউজার যেকোনটি সিলেক্ট করে দিতে পারে (তারিখ নির্বাচনের জন্য এরুপ ড্রপডাউন লাগতে পারে), এখানে লুপ ব্যবহার করতে হবে।

- ডেটাবেসে হাজার হাজার ডেটা আছে, কোয়েরি করে একটা লুপে ফেলে দিলেই একটার পর একটা ডেটা আসতে থাকবে।ইত্যাদি।

পিএইচপিতে কয়েক ধরনের লুপ আছে যেমন

ফর লুপ (**for** loop)

হোয়াইল লুপ (**while** loop)

ফরইচ লুপ (**foreach** loop)

ডু হোয়াইল লুপ (**do**...**while** loop)

মুলত সবচেয়ে বেশি লাগে ফর লুপ, ফরইচ এবং এরপর হোয়াইল লুপ। ডু হোয়াইল কম লাগে।

**PHP For Loop:**

The for loop is used when you know in advance how many times the script should run.

for (*init counter; test counter; increment counter*) {  
  *code to be executed;*  
}

Parameters:

*init counter*: Initialize the loop counter value

*test counter*: Evaluated for each loop iteration. If it evaluates to TRUE, the loop continues. If it evaluates to FALSE, the loop ends.

*increment counter*: Increases the loop counter value

এই লুপ টি একটু জটিল। ফর লুপ এর কাজ while লুপ দিয়েও হয়, তবে কিছু সুবিধার কারনে এটি ব্যাহৃত হয়।

**সংকেত**

1.for (expr1; expr2; expr3)

2.{

3.code to be executed;

4.}

for লুপ স্টেটমেন্ট ব্রাকেটের ভিতর ৩টি expression নেয় যেগুলো সেমিক্লোন দ্বারা বিভক্ত হয়। ১ম টি assignment statement(loop control variable), প্রথমবার লুপটি পূনরাবৃত্তির আগেই এটা একবার execution হয়। ২য় টি Boolean expression  যেটা প্রতিবার পূনরাবৃত্তির আগেই একবার এর মান নির্নীত(evaluate)হয়, এই মান true হলে পূনরাবৃত্তি চলবে আর যদি false return  করে তাহলে পূনরাবৃত্তি/লুপিং বন্ধ হয়ে যাবে। ৩য় টি দিয়ে loop control variable এর মান বাড়াতে বা কমাতে ব্যাবহৃত হয়। নিচের উদাহরনটি দেখুন লুপটি শুরু হয়েছে $i=1 দিয়ে এবং এটি চলবে যতক্ষন $i এর মান ৫ এর চেয়ে ছোট বা সমান হয়। আর $i এর মান ১ করে বাড়বে।

**1.<?php**

**2.for ($i=1; $i<=5; $i++){**

**3.echo "The number is " . $i . "<br />";**

**4.}**

**5.?>**

আউটপুট

The number is 1  
The number is 2  
The number is 3  
The number is 4  
The number is 5

**The PHP foreach Loop:**

The foreach loop works only on arrays, and is used to loop through each key/value pair in an array.

**Syntax**

foreach ($*array*as$*value*) {  
    *code to be executed;*  
}

For every loop iteration, the value of the current array element is assigned to $value and the array pointer is moved by one, until it reaches the last array element.

The following example demonstrates a loop that will output the values of the given array ($colors):

Example

<?php   
$colors = array("red", "green", "blue", "yellow");   
  
foreach ($colors as $value) {  
    echo "$value <br>";  
}  
?>

**PHP While Loop:**

The while loop executes a block of code as long as the specified condition is true.

Syntax

while (condition is true) {

code to be executed;

}

যদি condition false হয় তখন সেকেন্ড ব্রাকেটের ভিতর (কোডব্লকটি) কোডটি এড়িয়ে যায় (ব্রাকেট না বলে curly braces বলা উচিৎ)। যখন শেষের ব্রাকেটটি আসবে তখন condition আবার চেক করবে, true হলে কোড আবার execute করবে। condition এ যতবার দেয়া আছে ততবার এভাবে চলবেই। যেমন:

1.<?php

2.$i=1;

3.while($i<6){

4.echo 'A counter start from '. $i . '<br/>';

5.$i++;

6.}

7.?>

আউটপুট

A counter start from 1  
A counter start from 2  
A counter start from 3  
A counter start from 4  
A counter start from 5

ব্যাখ্যা: প্রথমে $i এর মান ১ দিয়ে শুরু হয়েছে। এরপর while লুপের ভিতর শর্তটি ($i<6) চেক করবে $i এর মান কত?

১ম বার $i এর মান ১ তার মানে ১ এখানে ৬ এর চেয়ে ছোট তাই এখন কোডব্লকে ঢুকবে এবং সেখানকার echo স্টেটমেন্ট বা যে কোডই থাকুন একবার এক্সিকিউট করবে। এবং $i++ এর কারনে $i এর মান ১ বাড়াবে।

২য় বার, তাহলে এবার $i এর মান পাবে ২ এবং এটাও ৬ এর চেয়ে ছোট ফলে শর্তটি ($i<6) true রিটার্ন করবে এবং লুপটি আবার একবার কোডব্লকটি এক্সিকিউট করবে। যেহেতু লুপের ভিতর $i++ আছে তাই আবার $i এর ১ বাড়াবে সুতরাং $i এর মান এই লুপ শেষে হল ৩। কেননা প্রথমবার ১ বাড়িয়ে ২ হয়েছিল।

এভাবে ৫ বার লুপটি পূনরাবৃত্তি হবে। যখনি $i এর মান বাড়তে বাড়তে ৬ হয়ে যাবে তখন শর্তটি ($i<6) আর মিলবেনা কেননা ৬ তো ৬ এর ছোট নয় বরং সমান সুতরাং শর্তটি false রিটার্ন করবে।

এবার প্রথমে $i = 6 দিলে কি আউটপুট আসবে? কিছুই আসবেনা কারন তখন ($i<6) এই শর্তের এর সাথে মিলবেনা তাই কোড execute হবেনা। অর্থাৎ এখন $i (বা ৬) ৬ এর চেয়ে ছোট নয়। প্রাকটিস করে দেখুন