**BÀI 2: CÁC KIỂU DỮ LIỆU – CẤU TRÚC RẼ NHÁNH – VÒNG LẶP**

1. **Kiểu dữ liệu:**
2. Mô tả:

|  |  |
| --- | --- |
| **Kiểu dữ liệu** | **Mô tả** |
| Boolean | True/False |
| Interger | Kiểu số nguyên. Giá trị có thể là số trong hệ thập phân, thập lục, bát phân |
| Float/Double | Kiểu số thực |
| String | Kiểu chuỗi, Kí tự. Mỗi kí tự = 1 byte Chuỗi đặt trong nháy đơn ‘’ hoặc nháy kép “” |
| Array | Kiểu mảng có các phần tử |
| JSON | Kiểu json |
| Object | Kiểu đối tượng của 1 lớp |

Ví dụ:

* Kiểu array:

|  |
| --- |
| *<?php*  *$array = array(1,2,3,4);*  *Print\_r($array);*  *?>* |

* Kiểu objects:

|  |
| --- |
| <?php  class a{  function chao\_a(){  echo “Xin chào a”;  }  }  $b = new a();  $b->chao\_a(); *// kết quả : Xin chào a*  ?> |

1. Chuyển đổi kiểu dữ liệu:

|  |
| --- |
| <?php  $don\_gia = 5000; //kiểu int  $so\_luong = 100;  $thanh\_tien = (double)($so\_luong\*$don\_gia);  ?> |

1. **Biến và cách gán giá trị cho biến:** **biến** là một ô nhớ trong bộ nhớ biểu diễn thông tin thực tế trong chương trình
2. Khai báo biến:

* Bắt đầu bằng **$** theo sau là tên biến
* Nguyên tắt đặt tên: bắt đầu bằng 1 kí tự hoặc một dấu \_

**Chú ý:**

* PHP không yêu cầu phải khai báo biến trước khi sử dụng, tuy nhiên nên khai báo và khởi tạo giá trị ban đầu cho biến
* Không khai báo tên biến trùng với tên hàm
* Không nên bắt đầu bằng số
* Có phân biệt hoa thường

Ví dụ:

1. Gán giá trị cho biến:

---------------------------------------

1. **Hằng:**
2. Khai báo:

**Define(‘ten\_hang’, gia\_tri);**

1. Sử dụng:

Ví dụ:

|  |
| --- |
| <?php  Define(‘pi’, 3.14);  $r = 10;  $s = pi \* $r\*$r; // pi\*pow($r,2);  $c = 2\*pi\*$r;  ?> |

* Khác nhau giữa hàng và biến:
* Tên hằng ko có $
* Hằng chỉ có thể khai báo bằng define()
* Không khai báo lại hằng khi đã thiết lập

1. **Các toán tử**
2. Toán tử số học:

+ - \* / %

***Ví dụ:***

1. Toán tử nối chuỗi:

Dùng . để nối hai chuỗi

***Ví dụ:***

Chú ý: Kết quả khi dùng toán tử nối chuỗi là một chuỗi

1. Toán tử gán kết hợp:

|  |  |  |
| --- | --- | --- |
| **Kí hiệu** | **Tên** | **Sử dụng** |
| += | Cộng bằng | $a+=$b ; $a = $a+$b |
| ++ | Tăng 1 đơn vị | $a++; $a = $a+1 |
| -= | Trừ bằng | $a-=$b ; $a = $a-$b |
| -- | Giảm 1 đơn vị | $a--; $a = $a-1 |
| \*= | Nhân bằng | $a-\*=$b ; $a = $a\*$b |
| /= | Chia bằng | $a/=$b ; $a = $a/$b |
| %= | Dư bằng | $a%=$b ; $a = $a%$b |
| .= | Nối bằng | $a.=$b ; $a = $a.$b |

Ví dụ:

1. Toán tử so sánh:

|  |  |
| --- | --- |
| **Kí hiệu** | **Tên/ ý nghĩa** |
| == | So sánh bằng |
| === | So sánh bằng tuyệt đối |
| !=, <> | Khác, Không bằng |
| !== | Khác tuyệt đối |
| < | Bé hơn |
| <= | Bé hơn hoặc bằng |
| > | Lớn hơn |
| >= | Lớn hơn hoặc bằng |

Ví dụ:

1. Toán tử luận lý:

|  |  |  |
| --- | --- | --- |
| **Kí hiệu** | **Tên** | **Ý nghĩa** |
| ! | Not | Toán tử phủ định |
| And, && | And | Đúng khi hai biểu thức có giá trị true |
| Or, || | Or | Đúng khi một trong hai biểu thức có giá trị true, hoặc cả hai cũng có giá trị true |

Ví dụ 1: Kiểm tra tam giác và xác định loại tam giác (dùng &&, ||)

Hướng dẫn: Hình được xác định là tam giác khi tổng 2 cạnh lớn hơn 1 cạnh

Ví dụ 2: dùng !

a =2, b=1

if(!(a>b)) echo a ko lớn hơn b

else echo a lơn hơn b

1. **Các hàm kiểm tra giá trị của biến**
2. Kiểm tra tồn tại **isset()**

Ví dụ:

If(isset($\_GET[‘a’])){ echo $\_GET[‘a’];}

Else{ echo “không có a”}

Lưu ý: dùng var\_dump() để kiểm tra kết quả của isset(): var\_dump(isset($a))

1. Kiểm tra giá trị rỗng **empty()**

Dùng để kiểm tra biến có giá trị rỗng hay không

Biến có giá trị NULL, 0 hoặc chuỗi rỗng, kết quả trả về true, ngược lại trả về false

Ngược lại với hàm isset()

Ví dụ:

|  |
| --- |
| If(empty($a)){  Echo ‘Vui lòng nhập số a’;}  Else{  Echo “a = $a”;  } |

1. Kiểm tra trị kiểu số **is\_numeric():**

Ví dụ:

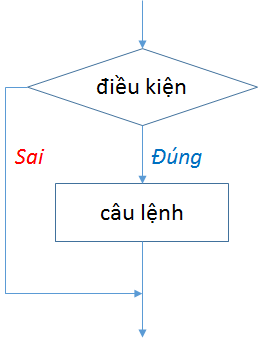
|  |
| --- |
| If(is\_numeric($so\_luong)){  Echo $so\_luong;  }  Else{  Echo ‘số lượng phải là kiểu số’;  } |

1. Kiểm tra kiểu dữ liệu của biến **is\_int(), is\_long(), is\_string(), is\_double():**
2. Xác định kiểu dữ liệu của biến **gettype()**

Ví du: $a = ‘2121’;

$b = 1212;

$c = array(1,2,3);

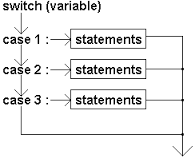
1. **Cấu trúc điều khiển**
2. Cấu trúc rẽ nhánh **if(){}else{}**

Ví dụ: so sánh hai số:

1. Toán tử **?:**

Ví du: so sánh 2 số a>b?a:b

1. Cấu trúc lựa chọn **switch….case**

ví dụ: đổi số thành chữ

|  |
| --- |
| $so = $\_POST[‘so’];  Switch($so){  Case 1: $chu = ‘một’; break;  Case 2: $chu = ‘hai’; break;  …  Default: $chu = ‘đây ko phải là số’;  }  Echo $chu; |

Bài tập : In câu chúc theo thứ trong tuần:

HD: switch($thu){

Case ‘thứ hai’:

Case ‘thứ ba’:

…

Echo ‘chúc ngày làm việc vui vẻ’;

Break;

Case ‘Thứ bảy’:

Case ‘Chủ nhật’:

Echo ‘cuối tuần vui vẻ’;

Break;

}

1. Cấu trúc lặp: for/foreach
2. **For**

|  |
| --- |
| For($i = <giá trị khởi đầu>; <điều kiện>; <giá trị lặp>){  } |

Ví dụ: in số từ 1->10

Bài tập: tính tổng từ 1->10

Vẽ bàn cờ

1. **foreach()**

|  |
| --- |
| foreach($tenmang as $giatri){  } |

Ví du: duyệt mảng và in giá trị trong mảng

Bài tập:duyệt mảng và in nội dung của mảng

HD:

foreach($mang as $key=>$value){

}

Bài tập: in bàn cờ vua

1. cấu trúc **while**

|  |
| --- |
| while(<điều kiện>)  {  <công việc>  } |

Ví dụ: tính tổng các số nguyên dương < 10 // kết quá 45

1. cấu trúc **do…while**

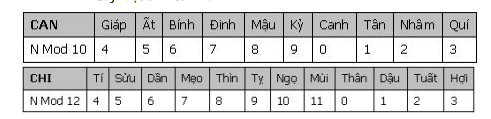
|  |
| --- |
| do{  <công việc>  }  While(<điều kiện>) |

Ví dụ: tính tổng các số nguyên dương < 10

* Bài tập:

1. Nhập năm dương lịch in ra năm âm lịch

Qui tắc xử lý : Tính can và chi theo quy luật tính can – chi   
Can = năm %10   
Chi = năm % 12   
Lấy can ghép với chi để có năm âm lịch Quy luật tính can – chi



1. Cho một mảng danh sách sản phẩm

$mangsp = array(

“1”=>array(“TenSP”=>”IPhone 5”,”Hinh”=>”iphone5.jpg”,”Gia”=>3000),

“2”=>array(“TenSP”=>”IPhone 6”,”Hinh”=>”iphone6.jpg”,”Gia”=>4000),

“3”=>array(“TenSP”=>”IPhone 7”,”Hinh”=>”iphone7.jpg”,”Gia”=>5000),

“4”=>array(“TenSP”=>”IPhone 8”,”Hinh”=>”iphone8.jpg”,”Gia”=>6000),

“5”=>array(“TenSP”=>”IPhone 9”,”Hinh”=>”iphone9.jpg”,”Gia”=>7000)

)

Mảng trên là 1 mảng chứa nhiều sản phẩm, mỗi sản phẩm lại là 1 mảng chứa thông tin sản phẩm  
Code gợi ý

Foreach($mangsp as $key=>$val)

{

//$val hiện giờ là 1 mảng thông tin, để xuất thông tin từ mảng $val, ta gọi $val[“TenSP”], $val[“Hinh”], $val[“Gia”]

}

Kết quả như sau:

