



**KHOA
PHẠM
.VN**

TRUNG TÂM ĐÀO TẠO TIN HỌC KHOA PHẠM

Website: <http://khoapham.vn>

Địa chỉ: 90 Lê Thị Riêng, P.Bến Thành, Q.1, TP.HCM

Điện thoại: 0966 908 907 - 094 276 4080

Facebook: <http://facebook.com/khoapham.vn>

KHOA PHAM TRAINING

BÀI 9: CÁC KIỂU DỮ LIỆU – CẤU TRÚC RỄ NHÁNH – VÒNG LẶP

I. Kiểu dữ liệu:

A. Mô tả:

Kiểu dữ liệu	Mô tả
Boolean	True/False
Integer	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);
?>
```



**KHOA
PHẠM
.VN**

TRUNG TÂM ĐÀO TẠO TIN HỌC KHOA PHẠM

Website: <http://khoapham.vn>

Địa chỉ: 90 Lê Thị Riêng, P.Bến Thành, Q.1, TP.HCM

Điện thoại: 0966 908 907 - 094 276 4080

Facebook: <http://facebook.com/khoapham.vn>

KHOA PHAM TRAINING

- Kiểu objects:

```
<?php

class a{

    function chao_a(){

        echo "Xin chào a";

    }

}

$b = new a();

$b->chao_a(); // kết quả : Xin chào a

?>
```

B. Chuyển đổi kiểu dữ liệu:

```
<?php

$don_gia = 5000; //kiểu int

$so_luong = 100;

$thanh_tien = (double)($so_luong*$don_gia);

?>
```

II. 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

a. Khai báo biến:

- Bắt đầu bằng \$ theo sau là tên biến
- Nguyên tắc 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:

b. Gán giá trị cho biến:

III. Hằng:

a. Khai báo:

Define('ten_hang', gia_tri);

b. 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

IV. Các toán tử

a. Toán tử số học:

+ - * / %

Ví dụ:

b. Toán tử nối chuỗi:

Dùng . để nối hai chuỗi

Ví dụ:

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

c. 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:

d. 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:

e. 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 lớn hơn b
```

V. Các hàm kiểm tra giá trị của biến

1. Kiểm tra tồn tại **isset()**

Ví dụ:

```
If(isset($_POST['a'])){ echo $_ POST ['a'];}  
Else{ echo "không có a"}
```

Lưu ý: dùng `var_dump()` để kiểm tra kết quả của `isset()`: `var_dump(isset($a))`

2. 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";
```

}

3. 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';
}
```

4. Kiểm tra kiểu dữ liệu của biến **is_int()**, **is_long()**, **is_string()**, **is_double()**:

5. Xác định kiểu dữ liệu của biến **gettype()**

Ví dụ: \$a = '2121';

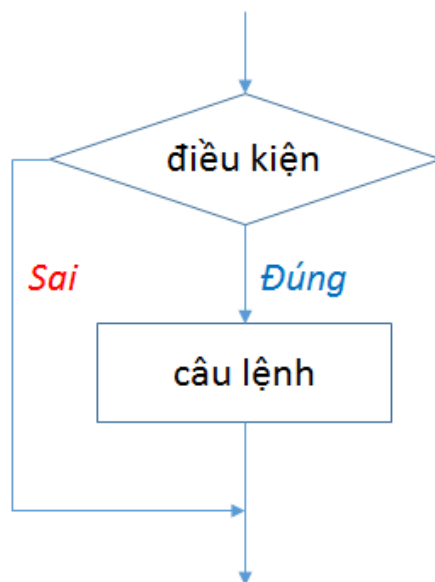
\$b = 1212;

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

VI. Cấu trúc điều khiển



1. Cấu trúc rẽ nhánh **if(){}else{}**

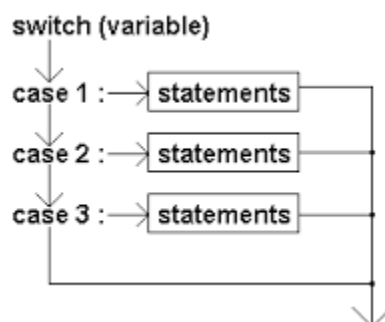


Ví dụ: so sánh hai số:

2. Toán tử **?:**

Ví dụ: so sánh 2 số $a > b$? $a : b$

3. 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;  
}
```

4. Cấu trúc lặp: for/foreach

a. 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

b. foreach()

```
foreach($tenmang as $giatri){  
  
}
```

Ví dụ: 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

5. cấu trúc while

```
while(<điều kiện>)  
  
{  
  
<công việc>  
  
}
```

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

6. cấu trúc do...while



do{

<công việc>

}

While(<điều kiện>)

Ví dụ: tính tổng các số nguyên dương < 10

- Bài tập:

- 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

CAN	Giáp	Ất	Bính	Đinh	Mậu	Kỷ	Canh	Tân	Nhâm	Quý
N Mod 10	4	5	6	7	8	9	0	1	2	3

CHI	Tí	Sửu	Dần	Mẹo	Thìn	Tỵ	Ngọ	Mùi	Thân	Dậu	Tuất	Hợi
N Mod 12	4	5	6	7	8	9	10	11	0	1	2	3