câu lệnh rẽ nhánh (đ kiện)

t, if - elseif - else

t, switch - case

>	!=	} sosánh
>=	==	} gtri
<	===	} ss gtri
<=	!==	} + kiểu data

① if (đk) {   toán tử sosanh & bitwise
}

bool true

if ($a === true) {

//...

}

}

0 rõ ràng datatype

"1" = 1 = true

"" = "0" = 0 = false

① toán tử ssánh

② toán tử bit vxlice $\left\{\begin{array}{l}\text{88} \\ \text{||}\end{array}\right.$ $\left|\begin{array}{l}\$a \\ \$b \\ \$c\end{array}\right.$

$\overset{true}{\overset{\text{''}}{if}} \left(\overset{true}{\overset{\text{''}}{\text{đk1}}} \quad 88 \quad \text{đk2} \right) \{$ tạo thành Δ khi nào?

ss! $///\Lambda$

$if (a+b) > c \quad 88 \quad a+c > b \quad 88 \quad b+c > a)$

$\times \left| if (a+b) < c \quad || \quad a+c > b \quad || \quad b+c > a) \right.$

hình thành Δ

b

phép và: tất cả các đk trong biểu thức bltxise phải trả về True (&&)

phép hoặc (||): chỉ cần 1 trong các biểu thức của bltxrice = True thì bthức vẫn đúng.

① if (đ/(c)đ
    câu lệnh
  }

② if (đ/(c1) {
   code 1
  } else if (đ/(c2) {
   code 2
  }

③ if (đ/(c1) {
     ①
  } else if (đ/(c2) {
     ②
  } else {
     ②
  }

④ } if (đ/(c1) {
    } else if (đ/(c2) {
    } else if (đ/(cn) {
    } else {
    }

⑤ biểu thức 3 ngôi

```
$a = rand(1, 2);

$text = $a % 2 == 1 ? "lẻ" : "chẵn";
```

```
if($a % 2 == 1){
    $text = "lẻ";
} else {
    $text = "chẵn";
}
```

```
for ($i = 0; $i < 100; $i++) {
    if (đk) {
        break;
    }
}

$cart[];    index

$sp1 = 1, A
$sp2 = 2, B
$sp3 = 3, C
```

flag

① : khi ta xem sp1 đó có trong giỏ hàng
chưa

```
$flag = -1;
for ($i = 0; $i < count($cart); $i++) {
    if ($cart[$i]["id"] == $id) {
        $flag = $i;
        break;
    }
}
```

```php
if ($flag == -1) => $item = {
                                    "id" => 1,
                                    "name" => A,
                                    "quantity" => 1
$cart[] = $item;

else {
    $cart[$flag]{"quantity"}++;  };
}
```

* switch - case ' [if ... else if ... else .

so sánh bằng .

switch (gtri. cần so sánh - biến) { ①

case "x".
    # 1
    break;
case "y".
    # 2
    break;
    ...
}

so sánh gtri (1) với cac gtri
đc viết trong case (x, y)

nếu (1) == "x" ⇒ các câu lệnh

(1) == "y" ⇒ #
    # 1
    sẽ chay

default : x & y đã ọ tìm
    # 3
    break; ⇒ # 3 sẽ chay

* Vòng lặp : thực hiện 1 đoạn code n lần với te
y' cầu phải tìm đk, khi đk ọ còn đúng ⇒ vòng lặp dừng

while (đk) {

do {

} while (đk)

for ($i = 0 ; $i < 100 ; $i++) {

for ($i = 100 ; $i > 0 ; $i--) {

100
0→99

100

100→1

}

}

tìm ra tất cả các số ng tố năm trong khoảng 0→1000

for 2 → 1000 →$i
$cout = false;
for 2 → i   for($j=2 ; $j < $i:$j ++) {
i f($i % $j == 0) {
o phải số ng tố . $count = true.

}

if($count == true) ⇒ số $i o phải số ng tố

== false ⇒ số $i là số ng tố .

```php
$arr = [1, 2, 5, 4, 9, 12, 11];
$arr[0];

$arr[3];

for($i=0; $i < count($arr); $i++){
    echo $arr[$i];
    echo "<br>";
}
```

① Tính tổng tất cả các ptử là số lẻ trong mảng arr.

② Sắp xếp mảng theo thứ tự tăng dần của ptử;
(sắp xếp nổi bọt)   các ptử.

```
$arr = [ 3, 4, 1, 7, 6, 5, 2 ];
       0  1  2  3  4  5  6

for( $i = 0; $i < count( $arr ); $i++ ){
                        7

    for( $j = $i+1; $j < count ($arr); $i++ )}
        if( $arr[$i] > $arr[$j] )}
            $tmp = $arr[$i];
            $arr[$i] = $arr[$j];
            $arr[$j] = $tmp;

                    sắp xếp nổi bọt
    }

}               => thuật toán
```

$i = 0 => 3

=> $j = 1 => 2

$arr[$i] > $arr[$j]

=> đổi chỗ $i $j

```
3 4 1 7 6 5 2.
1 4 3 7 6 5 2.
  3 4 7 6 5 2
  2 4 7 6 5 3
    3 7 6 5 4
```