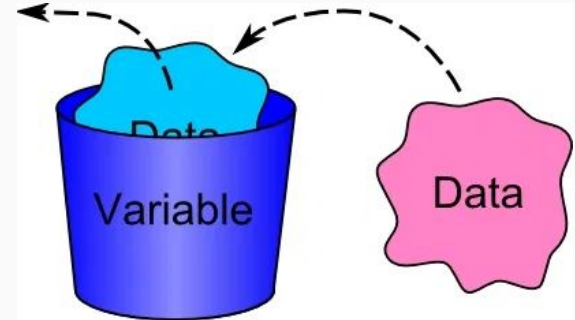


Programación Web

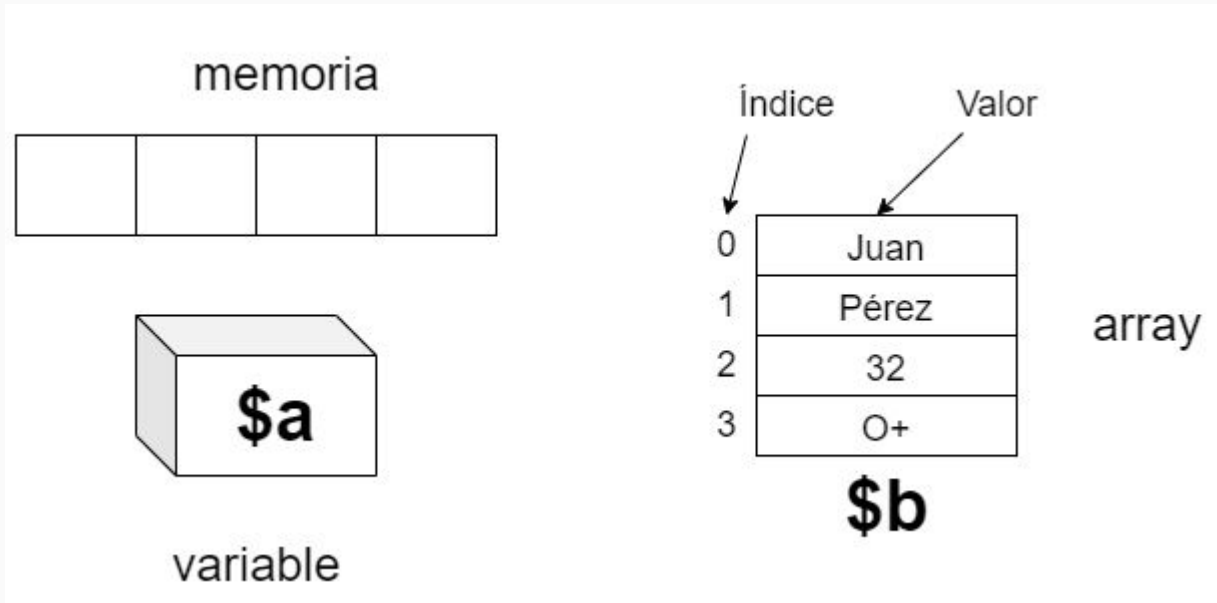
Sintaxis PHP

Variable

- Una variable, es una posición de memoria en la cual se almacena temporalmente un dato; a esta posición de memoria le ponemos un nombre para poder referenciarla a lo largo de nuestro código.
- Por ser un almacenamiento temporal, cada vez que le asignamos un valor, estamos sobrescribiendo el valor anterior.



Variable - Array



PHP - Sintaxis

- Es un lenguaje “**case sensitive**” es decir sensible a mayúsculas y minúsculas.
- PHP es un lenguaje que deriva del lenguaje C, por lo tanto **cada línea debe terminar en punto y coma.**
- Las variables comienzan con el símbolo de dólar **\$** y el primer carácter del nombre no puede ser un número.
- En PHP, **no es necesario declarar una variable** antes de usarla.

PHP - Sintaxis

- Se utiliza “**echo**” para lanzar una impresión en pantalla.
- Para la impresión de texto se puede utilizar:
 - Comillas simples
 - Comillas dobles (permiten que una variable muestre su valor).
- Los comentarios pueden escribirse:
 - **//** Una sola línea
 - **#** Una sola línea
 - **/*** muchas líneas ***/**

Operadores

Operadores aritméticos

Operador	Nombre	Ejemplo	Descripción
+	Suma	$4 + 7$	Suma dos números
-	Resta	$9 - 2$	Resta dos números
*	Multiplicación	$6 * 3$	Multiplica dos números
/	División	$15 / 3$	Divide dos números
%	Módulo	$15 \% 4$	Devuelve el resto de dividir ambos números, en este ejemplo el resultado es 3
++	Suma 1	<code>\$a++</code>	Suma 1 al contenido de una variable
--	Resta 1	<code>\$a--</code>	Resta 1 al contenido de una variable

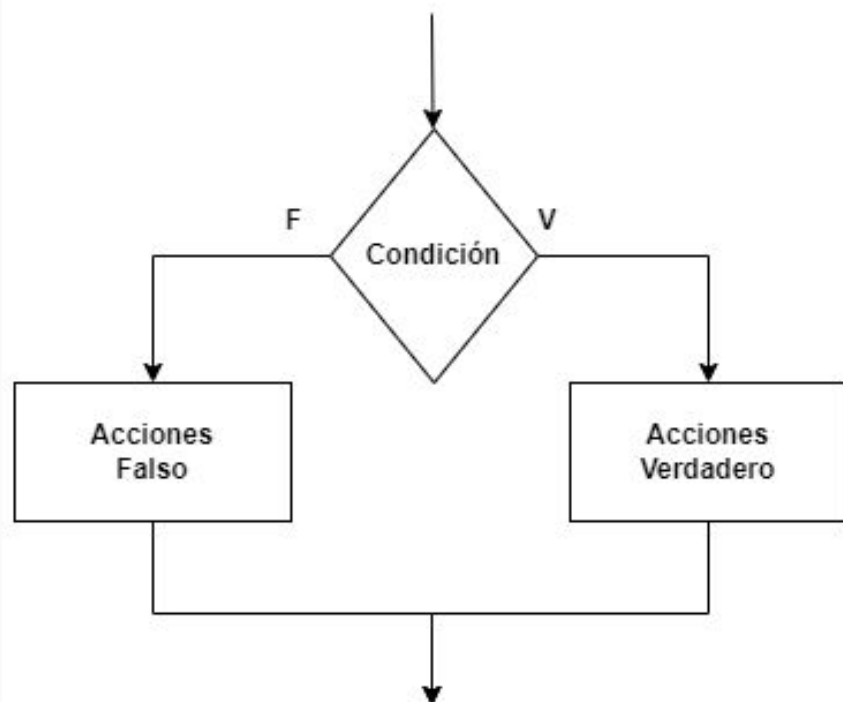
Operador	Nombre	Ejemplo	Devuelve verdadero cuando:
==	Igual	<code>\$a == \$b</code>	<code>\$a</code> es igual <code>\$b</code>
===	Idéntico	<code>\$a === \$b</code>	<code>\$a</code> es igual <code>\$b</code> y ambos son del mismo tipo
!=	Distinto	<code>\$a != \$b</code>	<code>\$a</code> es distinto <code>\$b</code>
!==	No idéntico	<code>\$a !== \$b</code>	<code>\$a</code> no es idéntico a <code>\$b</code>
<	Menor que	<code>\$a < \$b</code>	<code>\$a</code> es menor que <code>\$b</code>
>	Mayor que	<code>\$a > \$b</code>	<code>\$a</code> es mayor que <code>\$b</code>
<=	Menor o igual	<code>\$a <= \$b</code>	<code>\$a</code> es menor o igual que <code>\$b</code>
>=	Mayor o igual	<code>\$a >= \$b</code>	<code>\$a</code> es mayor o igual que <code>\$b</code>

Operadores lógicos

Operador	Nombre	Ejemplo	Devuelve cierto cuando:
&&	y	(10>3) && (5<8)	Devuelve verdadero cuando ambas condiciones son verdaderas
and	y	(10>3) and (5<8)	Devuelve verdadero cuando ambas condiciones son verdaderas
	o	(10>3) (5<8)	Devuelve verdadera cuando al menos una de las dos es verdadera
or	o	(10>3) or (5<8)	Devuelve verdadera cuando al menos una de las dos es verdadera
!	no	! (10>3)	Niega el valor de la expresión

CONDICIONALES

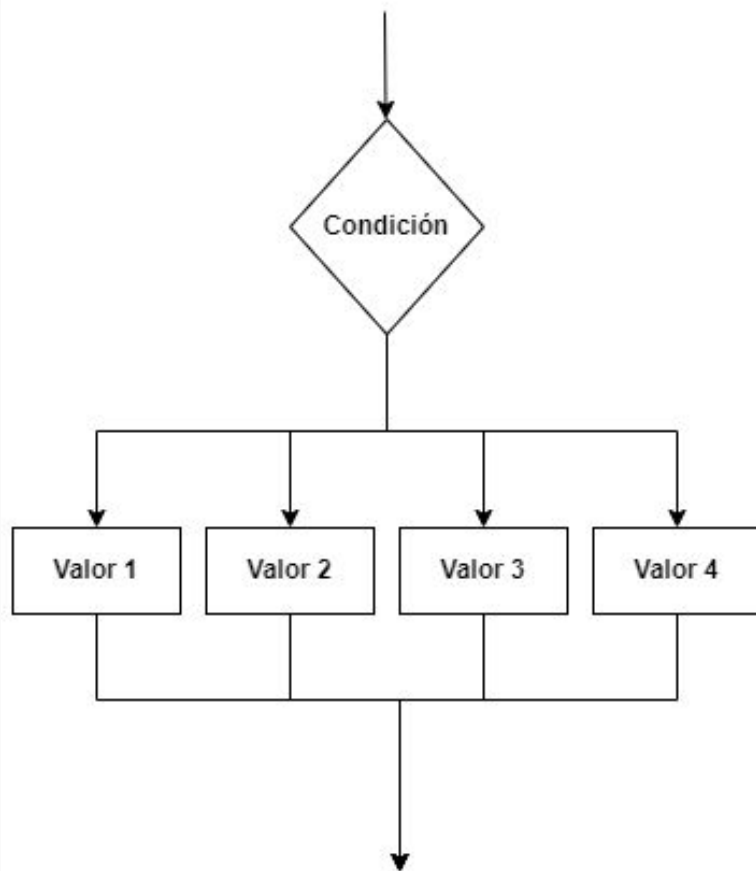
Condicional: IF



```
if(condicion){  
    acciones-verdadero  
}else{  
    acciones-falso  
}
```

(condicion ? verdadero : falso)

SWITCH

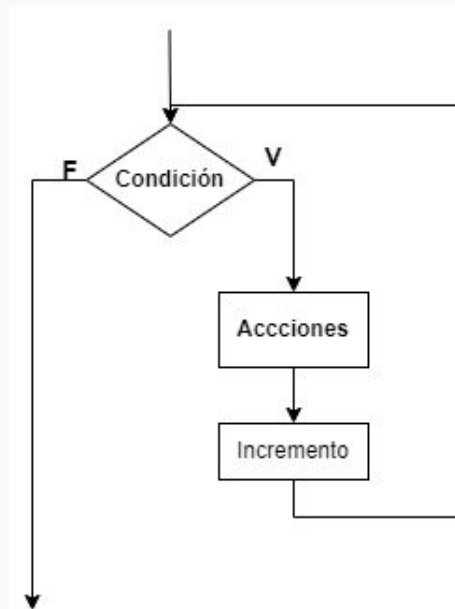


```
switch(condición){  
  case "valor 1":  
    acciones  
    break;  
  case "valor 2":  
    acciones  
    break;  
  case "valor 3":  
    acciones  
    break;  
  case "valor 4":  
    acciones  
    break;  
  default:  
    acciones  
}
```

BUCLES

FOR

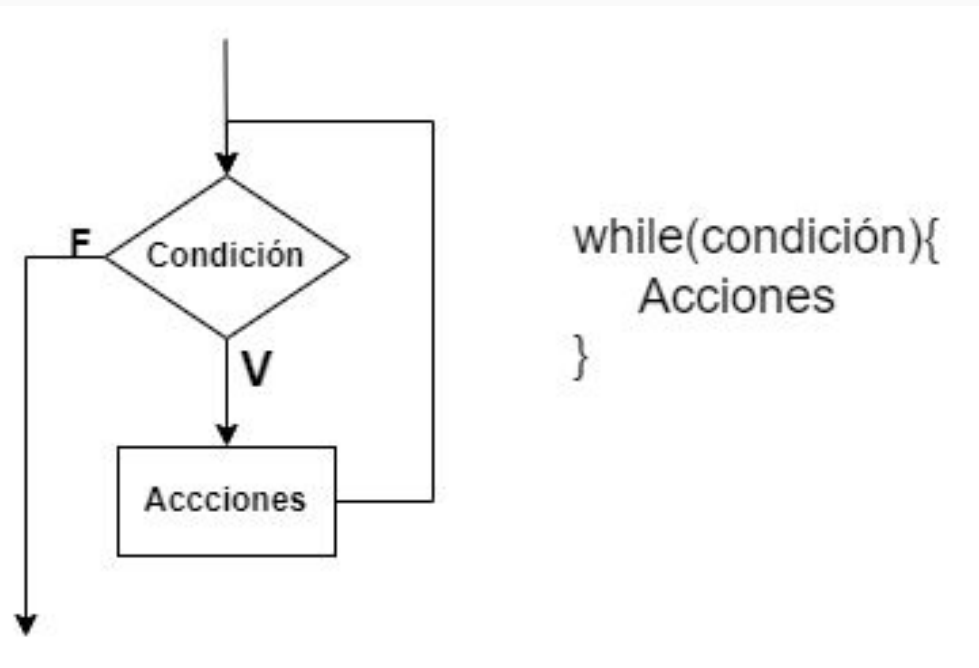
- Se conoce un valor inicial.
- En cada ciclo, se evalúa la condición.
- En cada ciclo, se aplica el incremento.



```
for (v_inicial ; condicion ; paso){  
    acciones;  
}
```

```
for ($m=1 ; $m<=10 ; $m++){  
    echo $m . "<br/>";  
}
```

While



Foreach (Recorrer arrays)

Gracias