

**PROGRAMA DE ESTUDIOS**

# **DESARROLLO DE SISTEMAS DE INFORMACIÓN**

**DESARROLLO DE  
BACKEND**

**Tema**

**APLICACIÓN DE LAS DIFERENTES  
ESTRUCTURAS CONDICIONALES  
( ELSEIF, SWITCH) EN EL LENGUAJE PHP.**

## APLICACIÓN DE LAS DIFERENTES ESTRUCTURAS

## CONDICIONALES (ELSEIF, SWITCH) EN EL LENGUAJE PHP.

En PHP, además de la estructura **if...else**, existen otras herramientas para manejar la lógica condicional, siendo **elseif** y **switch** las más comunes. Ambas sirven para tomar decisiones basadas en múltiples opciones, pero se aplican en diferentes contextos.

### 1. La Estructura elseif

El **elseif** es una combinación de **else** e **if** que te permite encadenar múltiples condiciones. PHP evalúa las condiciones de arriba abajo y, tan pronto como encuentra una que es verdadera, ejecuta el bloque de código asociado y salta el resto de la estructura.

#### Cuando usar elseif

El **elseif** es ideal cuando necesitas evaluar un rango de valores o condiciones que no están directamente relacionadas entre sí. Por ejemplo, para un sistema de calificaciones que asigna una letra según el rango de una puntuación.

#### Sintaxis y Ejemplo

La sintaxis de **elseif** se coloca entre un **if** inicial y un **else** final (que es opcional).

```
<?php
$puntuacion = 85;
$calificacion;

if ($puntuacion >= 90) {
    $calificacion = "A (Sobresaliente)";
} elseif ($puntuacion >= 80) {
    $calificacion = "B (Notable)";
} elseif ($puntuacion >= 70) {
    $calificacion = "C (Aprobado)";
} else {
    $calificacion = "D (Reprobado)";
}

echo "Tu calificación es: " . $calificacion;
// Salida: Tu calificación es: B (Notable)
?>
```

En este ejemplo, la puntuación de 85 es menor que 90, por lo que el primer **if** falla. Sin embargo, el primer **elseif** (**\$puntuacion >= 80**) es verdadero, así que se ejecuta ese bloque, se asigna la calificación "B" y el resto de la estructura se ignora.

## 2. La Estructura switch

La sentencia **switch** es una alternativa más limpia y eficiente al **if...elseif** cuando se necesita comparar una sola variable con múltiples valores exactos.

### Cuando usar switch

El **switch** es perfecto para situaciones donde la variable puede tener un conjunto limitado y conocido de valores, como el día de la semana, el rol de un usuario o el estado de un pedido. A diferencia de **elseif**, no es ideal para rangos de valores (>,<).

### Sintaxis y Ejemplo

La sintaxis de **switch** es más estructurada:

- **switch (expresión):** La expresión o variable a evaluar.
- **case valor:** El valor específico con el que se compara la expresión.
- **break:** Es fundamental. Evita que el programa siga ejecutando los siguientes case.
- **default:** Es opcional y se ejecuta si ningún case coincide.

```
<?php
$dia_de_la_semana = "Miercoles";
$mensaje;

switch ($dia_de_la_semana) {
    case "Lunes":
        $mensaje = "Empieza la semana con energía.";
        break;
    case "Viernes":
        $mensaje = "¡Casi es fin de semana!";
        break;
    case "Sábado":
    case "Domingo": // Puedes agrupar varios casos
        $mensaje = "¡Fin de semana!";
        break;
    default:
        $mensaje = "Es un día normal de la semana.";
        break;
}

echo $mensaje;
// Salida: Es un día normal de la semana.
?>
```

En este caso, PHP busca el **case** que coincide con el valor "**Miércoles**". Como no lo encuentra, ejecuta el bloque del **default**. Si **\$dia\_de\_la\_semana** fuera "**viernes**", ejecutaría ese **case** y se detendría por el **break**.

### ¿Cuándo elegir cuál?: elseif vs switch

Característica	elseif	switch
Uso	Evaluación de <b>múltiples condiciones</b> o rangos ( <code>&gt;=</code> , <code>&lt;</code> ).	Comparación de una variable con <b>múltiples valores exactos</b> ( <code>==</code> ).
Legibilidad	A veces puede ser difícil de leer si hay muchas condiciones complejas.	Generalmente más legible y ordenado para muchos casos.
Rendimiento	Ligeramente menos eficiente porque cada <code>if</code> se evalúa de forma independiente.	Suele ser más rápido porque solo se evalúa la expresión una vez al inicio.
<code>break</code>	No se necesita. La estructura se detiene automáticamente en la primera condición verdadera.	Indispensable para evitar la "caída" ( <code>fall-through</code> ) al siguiente <code>case</code> .

En conclusión, usa elseif cuando necesitas evaluar condiciones complejas o rangos de valores. Usa switch cuando tengas una variable con un conjunto discreto de valores posibles para una lógica más clara y eficiente.

#### Ejemplo combinado:

```
$opcion = "B";

if ($opcion == "A") {
    echo "Has elegido la opción A";
} elseif ($opcion == "B") {
    echo "Has elegido la opción B";
} else {
    echo "Opción no válida";
}

echo "<br>";

switch ($opcion) {
    case "A":
        echo "Switch: Opción A seleccionada";
        break;
    case "B":
        echo "Switch: Opción B seleccionada";
        break;
    default:
        echo "Switch: Opción no válida";
}
```

