

1. Indique el resultado que produce la ejecución del siguiente fragmento de código PHP:

```
<?php
define('UNA_CONSTANTE', 20);
echo "Valor de Una_Constante: " . Una_Constante . "\n";
echo "Valor de UNA_CONSTANTE: " . UNA_CONSTANTE . "\n";
define('UNA_CONSTANTE', 25);
echo "Valor de UNA_CONSTANTE modificada: " . UNA_CONSTANTE . "\n";
$valor = 2;
define('CONSTANTE2', "constante $valor: UNA_CONSTANTE");
echo "Valor de CONSTANTE2: " . CONSTANTE2 . "\n";
?>
```

2. Considere el siguiente fragmento de código:

```
<?php
$a = 1;
function Test() {
    echo $a;
}
Test();
?>
```

¿Cuál es la salida esperada de este programa? ¿Por qué?

Si en lugar del código anterior tenemos el siguiente:

```
<?php
$a = 1;
function Test() {
    global $a;
    echo $a;
}
Test();
?>
```

¿Hay alguna diferencia entre ambos fragmentos de código?. Justifique.

3. Indique el resultado de la ejecución del siguiente programa PHP:

```
<?php
function prueba1() {
    echo "\n--- En prueba1 ---\n";
    $valor1 = 'Definida en prueba1';
    echo "valor: $valor\n";
    echo "valor1: $valor1\n";
    prueba3($valor1);
}
```

```

function prueba2() {
    echo "\n--- En prueba2 ---\n";
    global $valor;
    $valor2 = 'Definida en prueba2';
    echo "valor: $valor\n";
    echo "valor1: $valor1\n";
    echo "valor2: $valor2\n";
}

function prueba3($valor) {
    echo "\n--- En prueba3 ---\n";
    echo "valor: $valor\n";
}

echo "\n--- Comienzo ---\n";
$valor = 'Definida fuera de una funcion';
echo "valor: $valor\n";
prueba3($valor);
prueba1();
prueba2();
?>

```

4. ¿Considera el segundo fragmento de código en el ejercicio 2 como una buena práctica de programación? ¿En qué casos consideraría útil el uso de una variable global? Justifique.

5. Indique el tipo de comparación que corresponde a cada operador y cuándo el resultado de la expresión es TRUE.

La primer expresión se da como ejemplo.

- \$a = \$b Asignación TRUE si \$b es TRUE
- \$a == \$b
- \$a === \$b
- \$a != \$b
- \$a <> \$b
- \$a !== \$b
- \$a < \$b
- \$a > \$b
- \$a <= \$b
- \$a >= \$b

6. Dado el siguiente fragmento de código, explique la salida y justifique.

Tenga en cuenta que la función var_dump() imprime el contenido de una variable.

```

<?php
var_dump(0 == "a");
var_dump("1" == "01");
switch ("a") {
case 0:
    echo "0";
    break;
case "a":
    echo "a";
    break;
}
?>

```

7. Indique cuáles de las siguientes expresiones tienen valor verdadero, y cuáles no son válidas. Justifique.

En todos los casos, debe tomarse:

```
$a = TRUE;  
$b = 3;  
$c = '0';  
define('D', 3.0);
```

- 1) \$a = \$b
- 2) \$b = \$a
- 3) 3 = TRUE
- 4) \$a == \$b
- 5) \$b == \$a
- 6) \$a === \$b
- 7) \$c == FALSE
- 8) \$c === FALSE
- 9) D = \$b
- 10) D == \$b
- 11) D === \$b
- 12) "D" == \$a
- 13) \$b == '3'
- 14) \$b == '3b'
- 15) \$b === '3b'
- 16) NULL > ''
- 17) is_numeric(1 + \$c)
- 18) is_numeric(\$c + 1)
- 19) is_string(\$c + 1)
- 20) is_bool(\$a = \$c)
- 21) is_bool(\$c = \$a)
- 22) is_real(\$c + D)

Luego, codifique un programa PHP para comprobar si los resultados son lo que esperaba.

8. Suponga que en los ejercicios de ordenamiento del práctico anterior, realizara una implementación con variables globales, en lugar de pasar los datos como parámetros. Además de desaprobar el parcial, ¿habría otra consecuencia?

9. Implemente una función que permita verificar si un string dado representa una fecha válida, retornando un valor booleano. Además del string, la función debe recibir dos parámetros enteros que indiquen el límite mínimo y máximo de los años permitidos.

Asuma que una fecha válida tiene el formato "dd/mm/aaaa" y que todos los meses tienen 31 días. Por ejemplo, "25/09/2007" es válida, pero "25/9/2007", "09/25/2007", "00/01/2007", "25/09/07" o "25-09-2007" no.

Se deben utilizar constantes en todos los lugares aplicables; por ejemplo, para el separador de los elementos de la fecha ('/').

Ayuda: la función intval(\$var) permite convertir un string en un entero.