

The background is a dark gray with white line art of circuit boards in the corners. The top-left and bottom-right corners feature more complex circuit patterns, while the top-right and bottom-left corners have simpler, more linear patterns.

# MANUAL PHP

## ejercicios básicos, funciones y arreglos

JUAN PABLO SAAVEDRA CHAMBO

# Ejercicios basicos

## 01. IMPRIMIR "HOLA MUNDO"

### INDEX

```
<?php
    include("../Libreria/HolaMundo.php")

    ?>
<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Hola Mundo</title>
</head>
<body>

</body>
```

### PHP

```
<?php
$saludo = "Hola Mundo";
var_dump($saludo);
?>
```

El código PHP asigna el valor "Hola Mundo" a la variable \$saludo y luego imprime el contenido de la variable usando var\_dump()

VARIABLES	TIPO
\$saludo	STRING

```
string(10) "Hola Mundo"
```

# Ejercicios basicos

## 2. IMPRIMIR LA SUMA DE DOS NÚMEROS

### INDEX

```
<?php
    include("../Libreria/SumaDeDosNumeros.php")

    ?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Suma De Dos Numero</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php
$numerouno = 9;
$numerodos = 15;
$resultado = $numerouno + $numerodos;
var_dump("El Resultado De La Suma Es:",$resultado);
?>
```

Este código calcula la suma de dos números y luego imprime un mensaje junto con el resultado de la suma.

VARIABLES	TIPO
\$numerouno	INT
\$numerodos	INT
\$resultado	INT

```
string(27) "El Resultado De La Suma Es:" int(24)
```

# Ejercicios basicos

## 3. IMPRIMIR LA RESTA, MULTIPLICACIÓN Y DIVISIÓN DE DOS NÚMEROS

### INDEX

```
<?php

    include("../Libreria/Operaciones.php")

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Operaciones</title>
</head>
<body>

</body>
```

### PHP

```
<?php
$numerouno = 40;
$numerosdos = 5;

$suma = $numerouno + $numerosdos;

$resta = $numerouno - $numerosdos;

$multiplicacion = $numerouno * $numerosdos;

$division = $numerouno / $numerosdos;

var_dump("El Resultado De La Suma Es:",+$suma);

echo("<br>");
echo("<br>");

var_dump("El Resultado De La Resta Es:",+$resta);

echo("<br>");
echo("<br>");

var_dump("El Resultado De La Multiplicacion Es:",+$multiplicacion);

echo("<br>");
echo("<br>");

var_dump("El Resultado De La Division Es:",+$division);
?>
```

# Ejercicios basicos

## 3. IMPRIMIR LA RESTA, MULTIPLICACIÓN Y DIVISIÓN DE DOS NÚMEROS

Este código calcula la suma de dos números y luego imprime un mensaje junto con el resultado de la suma.

VARIABLES	TIPO
\$numerouno	INT
\$numerodos	INT
\$suma	INT
\$resta	INT
\$multiplicacion	INT
\$division	INT

```
string(27) "El Resultado De La Suma Es:" int(45)
```

```
string(28) "El Resultado De La Resta Es:" int(35)
```

```
string(37) "El Resultado De La Multiplicacion Es:" int(200)
```

```
string(31) "El Resultado De La Division Es:" int(8)
```

# Ejercicios basicos

## 4. IMPRIMIR EL PORCENTAJE DE UN NÚMERO.

### INDEX

```
<?php
    include("../Libreria/PorcentajeDeUnNumero.php");

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Porcentaje De Un Numero</title>
</head>
<body>

</body>
```

### PHP

```
<?php
```

```
$numero = 25;
$porcentaje = $numero / 100;
```

```
var_dump("El Porcentaje Del Numero", $numero, "Es:", $porcentaje);
```

```
?>
```

Este calcula el porcentaje de un número dado. Inicialmente, se asigna el valor 25 a la variable \$numero. Luego, se calcula el porcentaje dividiendo este número entre 100 y se guarda en la variable \$porcentaje. Posteriormente, se utiliza la función var\_dump() para mostrar un mensaje que indica el valor original del número seguido del cálculo del porcentaje obtenido.

VARIABLES	TIPO
\$numero	INT
\$porcentaje	INT

```
string(24) "El Porcentaje Del Numero" int(25) string(3) "Es:" float(0.25)
```

# Ejercicios basicos

## 5. IMPRIMIR EL PROMEDIO DE 3 NOTAS

### INDEX

```
<?php

    include("../Libreria/PromedioDeTresNotas.php")

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Promedio De Tres Notas</title>
</head>
<body>

</body>
</html>
</body>

PHP

<?php
$nota1 = 4.5;
$nota2 = 3.5;
$nota3 = 3.0;

$suma = $nota1 + $nota2 + $nota3;

$promedio = $suma / 3;

var_dump("El Promedio De Las 3 Notas Es:",$promedio);

?>
```

Este código en PHP calcula el promedio de tres notas. Primero, se definen las variables \$nota1, \$nota2 y \$nota3, que representan las calificaciones de tres asignaturas. Luego, se suman estas calificaciones y se almacena el resultado en la variable \$suma. A continuación, se calcula el promedio dividiendo la suma de las calificaciones entre 3 y se guarda en la variable \$promedio. Finalmente, se utiliza la función var\_dump() para mostrar en la salida el mensaje

VARIABLES	TIPO
\$nota1	FLOAT
\$nota2	FLOAT
\$nota3	FLOAT
\$suma	INT
\$promedio	FLOAT

```
string(30) "El Promedio De Las 3 Notas Es:" float(3.6666666666666665)
```

# Ejercicios basicos

## 6. Imprimir El porcentaje de tres notas

### INDEX

```
<?php

    include("../Libreria/PorcentajeDeTresNotas.php");

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Porcentaje De Tres Notas</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php
$porcentaje1;
$porcentaje2;
$porcentaje3;

$nota1 = 4.3;
$nota2 = 3.5;
$nota3 = 3.2;

$porcentaje1 = $nota1 * 30 / 100;

$porcentaje2 = $nota2 * 30 / 100;

$porcentaje3 = $nota3 * 40 / 100;

var_dump("El Primer Porcentaje Es:",$porcentaje1);

echo("<br>");
echo("<br>");

var_dump("El Segundo Porcentaje Es:",$porcentaje2);

echo("<br>");
echo("<br>");

var_dump("El Tercer Porcentaje Es:",$porcentaje3);

echo("<br>");
echo("<br>");

$suma = $porcentaje1 + $porcentaje2 + $porcentaje3;

var_dump("La Suma De Los 3 Porcentajes Es:",$suma);

?>
```



# Ejercicios basicos

## 6. Imprimir El porcentaje de tres notas

Este código en PHP calcula los porcentajes correspondientes a tres notas, donde cada nota tiene un peso diferente en el cálculo del promedio final. Primero, se definen las variables \$porcentaje1, \$porcentaje2 y \$porcentaje3 para almacenar los porcentajes respectivos de las notas. Luego, se asignan los valores de las notas en las variables \$nota1, \$nota2 y \$nota3. A continuación, se calcula el porcentaje de cada nota multiplicando la nota por el peso asignado y dividiendo por 100.

Después de calcular cada porcentaje, se utiliza la función var\_dump() para mostrar en la salida el valor de cada porcentaje

VARIABLES	TIPO
\$porcentaje1	INT
\$porcentaje2	INT
\$porcentaje3	INT
\$nota1	INT
\$nota2	INT
\$nota3	INT

```
string(24) "El Primer Porcentaje Es:" float(1.29)
```

```
string(25) "El Segundo Porcentaje Es:" float(1.05)
```

```
string(24) "El Tercer Porcentaje Es:" float(1.28)
```

```
string(32) " La Suma De Los 3 Porcentajes Es:" float(3.62)
```

# Ejercicios basicos

## 7. Imprimir áreas figuras

### INDEX

```
<?php

    include("../Libreria/AreaFiguras.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Areas De Figuras</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php

$cuadrado;
$rectangulo;
$triangulo;

$lado = 14;
$base1 = 10;
$base2 = 5;
$base3 = 13;
$altura1 = 19;
$altura2 = 17;

$cuadrado = $lado * $lado;
$rectangulo = $base1 * $altura1;
$triangulo = $base2 * $altura2 / 2;

var_dump("El Area Del Cuadrado Es: ",+$cuadrado);
echo("</br>");
echo("</br>");

var_dump("El Area Del Rectangulo Es: ",+$rectangulo);
echo("</br>");
echo("</br>");

var_dump("El Area Del Triangulo Es: ",+$triangulo);
echo("</br>");
echo("</br>");

?>
```

# Ejercicios basicos

## 7. Imprimir áreas figuras

Este código en PHP calcula el área de un cuadrado, un rectángulo y un triángulo dados sus respectivos lados, bases y alturas, y luego muestra los resultados en la salida.

VARIABLES	TIPO
\$cuadrado	INT
\$rectangulo	INT
\$triangulo	FLOAT
\$lado	INT
\$base1	INT
\$base2	INT
\$base3	INT
\$altura1	INT
\$altura2	INT

```
string(25) "El Area Del Cuadrado Es: " int(196)
```

```
string(27) "El Area Del Rectangulo Es: " int(190)
```

```
string(26) "El Area Del Triangulo Es: " float(42.5)
```

# Ejercicios basicos

## 8. Imprimir Sueldo De Un Empleado

### INDEX

```
<?php

    include("../Libreria/SueldoEmpleado.php")

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Sueldo De Un Empleado</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php
$diasTrabajados = 13;
$valorDia = 50000;
$salario;
$salud;
$pension;
$arl;
$descuento;
$pagoTotal;

$salario = $diasTrabajados * $valorDia;
$salud = $salario * 0.12;
$pension = $salario * 0.16;
$arl = $salario * 0.052;
$descuento = $salud + $pension + $arl;
$pagoTotal = $salario - $descuento;

var_dump("La Salud Tiene Un Valor De: ",+$salud);
echo("</br>");
echo("</br>");

var_dump("La Pension Tiene Un Valor De: ",+$pension);
echo("</br>");
echo("</br>");

var_dump("El Arl Tiene Un Valor De: ",+$arl);
echo("</br>");
echo("</br>");

var_dump("El Sueldo Total Del Empleado Es: ",+$pagoTotal);
echo("</br>");
echo("</br>");

?>
```

# Ejercicios basicos

## 8. Imprimir Sueldo De Un Empleado

El código calcula el salario total de un empleado basado en el número de días trabajados y el valor del día, luego calcula los descuentos por salud, pensión y ARL, y finalmente muestra el salario neto del empleado.

VARIABLES	TIPO
\$diasTrabajados	INT
\$valorDia	INT
\$salario	INT
\$salud	INT
\$pension	INT
\$arl	INT
\$descuento	INT
\$pagoTotal	INT
\$altura2	INT

```
string(28) "La Salud Tiene Un Valor De: " int(78000)
```

```
string(30) "La Pensión Tiene Un Valor De: " int(104000)
```

```
string(26) "El Arl Tiene Un Valor De: " int(33800 )
```

```
string(33) "El Sueldo Total Del Empleado Es: " int(434200)
```

# CONDICIONALES

## 9. Imprimir si es mayor de edad

### INDEX

```
<?php

    include("../Libreria/MayorDeEdad.php")

    ?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Mayor De 18 Años</title>
</head>
<body>
<?php

var_dump($result);

?>
</body>
```

Este código en PHP verifica si la edad proporcionada es mayor o menor de 17 años y muestra un mensaje correspondiente.

VARIABLES	TIPO
\$edad	INT

### PHP

```
<?php

$edad = 16;

var_dump("Su Edad Es: ".$edad);
echo("</br>");
echo("</br>");

if($edad>17){
    $result = "Usted Es Mayor De Edad";
}else{
    $result = "Usted Es Menor De Edad";
}

?>
```

string(14) "Su Edad Es: 16"

string(22) "Usted Es Menor De Edad"

# CONDICIONALES

## 10. Imprimir el mayor de dos numeros

### INDEX

```
<?php

    include("../Libreria/MayorDeDosNumeros.php")

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Mayor De Dos Numeros</title>
</head>
<body>
<?php

var_dump($resul);

?>
</body>
</html>
```

### PHP

```
<?php

$num1 = 15;
$num2 = 20;

if($num2>$num1){
    $resul = "El Numero ".$num2." Es Mayor";
}else{
    $resul = "El Numero ".$num1." Es Menor";
}

?>
```

Este código en PHP compara dos números y determina cuál es el mayor de los dos, mostrando un mensaje indicando el resultado.

VARIABLES	TIPO
\$num1	INT
\$num2	INT

string(21) "El Numero 20 Es Mayor"



# CONDICIONALES

## 11. Imprimir Si Dos Números, Cual Es EL Mayor O Si Son Iguales

### INDEX

### PHP

```
<?php

    include("../Libreria/MayorOIgualDeDosNumeros.php")

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Mayor O Igual De Dos Numeros</title>
</head>
<body>
<?php

?>
</body>
</html>
```

```
<?php
$num1 = 25;
$num2 = 15;

var_dump("El Primer Numero Es: ".$num1);

echo("<br>");
echo("<br>");

var_dump("El Segundo Numero es: ".$num2);
echo("<br>");
echo("<br>");

if( $num1 == $num2){
    var_dump("Los Numeros Son Iguales");
}else{
    if( $num1 >$num2){
        var_dump("El numero ".$num1." es mayor");
    }else{
        var_dump("El numero ".$num2."es menor");
    }
}
```



# CONDICIONALES

## 11. Imprimir Si Dos Números, Cual Es EL Mayor O Si Son Iguales

Este código en PHP compara dos números y muestra un mensaje indicando el que es mayor, menor o si son iguales

VARIABLES	TIPO
\$num1	INT
\$num2	INT

```
string(23) "El Primer Numero Es: 25"
```

```
string(24) "El Segundo Numero es: 15"
```

```
string(21) "El numero 25 es mayor"
```

# CONDICIONALES

## 12. Calcular EL Area De Dos Cuadrados Y Imprimir Si Las Dos Areas Son Iguales O Cual INDEX Es La Mayor.

```
<?php

    include("../Libreria/AreaIgualOMayor.php")

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-
width, initial-scale=1.0">
    <title>Area Igual O Mayor</title>
</head>
<body>
</body>
</html>
```

<?php PHP

```
$area1;
$area2;
$area3;

$lado = 15;
$lado2 = 13;
$lado3 = 10;

$area1 = $lado * $lado;
$area2 = $lado2 * $lado2;
$area3 = $lado3 * $lado3;

if($area1 == $area2 && $area2 == $area3 && $area3 == $area1){
    var_dump($area1);
    var_dump($area2);
    var_dump($area3);
}else{

    if($area1 > $area2 && $area2 > $area3){

        var_dump($area1 . " Es El Area Mas Grande");

    }else{
        if($area1 < $area2 && $area2 < $area3){

            var_dump($area2 . " Es El Area Mas Grande ");

        }else{

            var_dump($area3 . " Es El Area Mas Grande");

        }
    }
}

?>
```

# CONDICIONALES

## 12. Calcular EL Area De Dos Cuadrados Y Imprimir Si Las Dos Areas Son Iguales O Cual Es La Mayor.

Este código en PHP calcula y compara las áreas de tres cuadrados diferentes, luego muestra cuál de ellos tiene el área más grande.

| VARIABLES | TIPO |
|-----------|------|
| \$area1   | INT  |
| \$area2   | INT  |
| \$area3   | INT  |
| \$lado1   | INT  |
| \$lado2   | INT  |
| \$lado3   | INT  |

```
string(25) "225 Es El Area Mas Grande"
```

# CONDICIONALES

## 13. Calcular E Imprimir El Promedio Y Si El Promedio Cumple Con La Mayoria De Edad

### INDEX

```
<?php

    include("../Libreria/PromedioEdad.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Promedio De 3 Edades</title>
</head>
<body>
</body>
</html>
```

### PHP

```
<?php

$edad1;
$edad2;
$edad3;

$prom;
$suma;
$anoNacio1 = 2000;
$anoNacio2 = 2006;
$anoNacio3 = 2009;
$anoActu = 2024;

$edad1 = $anoActu - $anoNacio1;
$edad2 = $anoActu - $anoNacio2;
$edad3 = $anoActu - $anoNacio3;

if ($edad1 > 17) {

    var_dump("Es mayor de edad\n");

    echo("<br>");
    echo("<br>");
```

```
    } else {

        var_dump("Es menor de edad\n");

        echo("<br>");
        echo("<br>");
    }
}
```

```
if ($edad2 > 17) {

    var_dump("Es mayor de edad\n");

    echo("<br>");
    echo("<br>");
```

```
} else {

    var_dump("Es menor de edad\n");

    echo("<br>");
    echo("<br>");
}
```

```
if ($edad3 > 17) {

    var_dump("Es mayor de edad\n");

    echo("<br>");
    echo("<br>");
```

```
} else {

    var_dump("Es menor de edad\n");

    echo("<br>");
    echo("<br>");
}
```

```
$suma = $edad1 + $edad2 + $edad3;
$prom = $suma / 3;
```

```
if ($prom > 17) {
    var_dump("El promedio de las tres edades es $prom y cumple con la mayoría de edad.\n");
} else {
    var_dump("El promedio no cumple con la mayoría de edad.\n");
}
?>
```

# CONDICIONALES

## 13. Calcular E Imprimir El Promedio Y Si El Promedio Cumple Con La Mayoria De Edad

Este código en PHP calcula la edad de tres personas a partir de sus años de nacimiento y determina si son mayores o menores de edad. Luego, calcula el promedio de las tres edades y verifica si este promedio cumple con la mayoría de edad.

VARIABLES	TIPO
\$edad1	INT
\$edad2	INT
\$edad3	INT
\$prom	INT
\$suma	INT
\$anoNacio1	INT
\$anoNacio2	INT
\$anoNacio3	INT
\$anoActu	INT

```
string(17) "Es mayor de edad "
```

```
string(17) "Es mayor de edad "
```

```
string(17) "Es menor de edad "
```

```
string(71) "El promedio de las tres edades es 19 y cumple con la mayoría de edad. "
```

# CONDICIONALES

## 14. Pago Total Del sueldo de una persona

### INDEX

```
<?php
```

```
include("../Libreria/PagoTotalSueldoPersona.php")
```

```
?>
```

```
<!DOCTYPE html>
```

```
<html Lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Sueldo Total De Una Persona</title>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```

### PHP

```
<?php
```

```
$salario = 1500000;
```

```
$dias = 43;
```

```
$valorDia;
```

```
$salud;
```

```
$pension;
```

```
$arl;
```

```
$transporte;
```

```
$salarioMin = 868800;
```

```
$deducible;
```

```
$valorDia = $salario / $dias;
```

```
$salario = $dias * $valorDia;
```

```
var_dump("El salario de la persona es $" . $salario . PHP_EOL);
```

```
echo("<br>");
```

```
echo("<br>");
```

```
if ($salarioMin * 2 < $salario) {
```

```
    $salario += 1;
```

```
}
```

```
$salud = $salario * 0.12;
```

```
$pension = $salario * 0.16;
```

```
$arl = $salario * 0.052;
```

```
// Impresión de los aportes
```

```
var_dump("El subsidio de salud de la persona es $" . $salud . PHP_EOL);
```

```
echo("<br>");
```

```
echo("<br>");
```

```
var_dump("El subsidio de la pensión de la persona es $" . $pension .
```

```
PHP_EOL);
```

```
echo("<br>");
```

```
echo("<br>");
```

```
var_dump("El subsidio de ARL de la persona es $" . $arl . PHP_EOL);
```

```
echo("<br>");
```

```
echo("<br>");
```

```
if ($salario >= $salarioMin * 4) {  
    $descuentoParafiscales = $salario * 0.04;
```

```
} else {
```

```
    $descuentoParafiscales = 0;
```

```
}
```

```
$deducible = $salud + $pension + $arl;
```

```
$salarioNeto = $salario - $deducible;
```

```
var_dump("El salario total de la persona es $" . $salarioNeto .  
PHP_EOL);
```

```
?>
```

# CONDICIONALES

## 14. Pago Total Del sueldo de una persona

Este código en PHP calcula el salario neto de una persona a partir de su salario mensual y la cantidad de días trabajados en el mes. También determina los descuentos por salud, pensión y ARL, así como el subsidio de transporte si aplica, para finalmente imprimir el salario total después de los descuentos.

VARIABLES	TIPO
\$salario	INT
\$dias	INT
\$valorDia	INT
\$salud	INT
\$pension	INT
\$arl	INT
\$transporte	INT
\$salarioMin	INT
\$deducible	INT

```
string(38) "El salario de la persona es $1500000 "
```

```
string(47) "El subsidio de salud de la persona es $180000 "
```

```
string(53) "El subsidio de la pensión de la persona es $240000 "
```

```
string(44) "El subsidio de ARL de la persona es $78000 "
```

```
string(44) "El salario total de la persona es $1002000 "
```

# CONDICIONALES

## 15. Porcentaje De 3 Notas y Decir cual es superior, buena o mala

### INDEX

```
<?php

    include("../Libreria/NotaPorcentajeCalificacion.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Porcentaje De 3 Notas Con Su Respectiva Calificacion</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php

$nota1 = 3.5;
$nota2 = 2.4;
$nota3 = 4.6;

$porc1;
$porc2;
$porc3;

$porc1 = $nota1 * (20 / 100);
$porc2 = $nota2 * (35 / 100);
$porc3 = $nota3 * (45 / 100);

var_dump("Porcentaje de la nota 1 es " . $porc1 . "\n");
echo("</br>");
echo("</br>");
```

```
var_dump("Porcentaje de la nota 2 es " . $porc2 . "\n");
echo("</br>");
echo("</br>");
```

```
var_dump("Porcentaje de la nota 3 es " . $porc3 . "\n");
echo("</br>");
echo("</br>");
```

```
$suma = $porc1 + $porc2 + $porc3;
```

```
var_dump("La suma de los porcentajes es " . $suma . "\n");
echo("</br>");
echo("</br>");
```

```
// Evaluar la nota final
if ($suma > 4.5) {
    var_dump("La nota es superior\n");
    echo("</br>");
    echo("</br>");
```

```
} else if ($suma > 3.5) {
    var_dump("La nota es buena\n");
    echo("</br>");
    echo("</br>");
```

```
} else if ($suma > 3) {
    var_dump("La nota es media\n");
    echo("</br>");
    echo("</br>");
```

```
} else {
    var_dump("La nota es mala\n");
    echo("</br>");
    echo("</br>");
}
```



# CONDICIONALES

## 16. Porcentaje De 3 Notas y Decir cual es superior, buena o mala

Este código en PHP calcula el porcentaje de tres notas y luego determina la categoría de la nota final en función de la suma de estos porcentajes, imprimiendo el resultado correspondiente.

VARIABLES	TIPO
\$nota1	FLOAT
\$nota2	FLOAT
\$nota3	FLOAT
\$porc1	FLOAT
\$porc2	FLOAT
\$porc3	FLOAT

```
string(31) "Porcentaje de la nota 1 es 0.7 "
```

```
string(32) "Porcentaje de la nota 2 es 0.84 "
```

```
string(32) "Porcentaje de la nota 3 es 2.07 "
```

```
string(35) "La suma de los porcentajes es 3.61 "
```

```
string(17) "La nota es buena "
```

# CICLOS

## 17. Contar del 1 al 5 WHILE

### INDEX

```
<?php

include("../Libreria/ContarNumeroWhile.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Contar Hasta el numero 5 while</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php

$contar;
$numero;

$contar = 0;
$numero = 5;

while( $contar < $numero){
  $contar = $contar+1;

  var_dump("Numero: ". $contar);
  echo("<br>");
  echo("<br>");
}

?>
```

Este código en PHP utiliza un bucle while para contar hasta un número específico, mostrando cada número en cada iteración del bucle.

VARIABLES	TIPO
\$contar	INT
\$numero	INT

cadena(9) "Numero: 1"

cadena(9) "Numero: 2"

cadena(9) "Numero: 3"

cadena(9) "Numero: 4"

cadena(9) "Numero: 5"

# CICLOS

## 18. Contar del 1 al 5 FOR

### INDEX

```
<?php

include("../Libreria/ContarNumeroFor.php")

?>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Contar del 1 al 5 For</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php

$contar;
$numero = 5;

for ($contar = 1; $contar <= $numero; $contar++){

    var_dump("Número " . $contar);
    echo("<br>");
    echo("<br>");
}

?>
```

Este código en PHP utiliza un bucle for para contar hasta un número específico, mostrando cada número en cada iteración del bucle.

VARIABLES	TIPO
\$contar	INT
\$numero	INT

cadena(9) "Número 1"

cadena(9) "Número 2"

cadena(9) "Número 3"

cadena(9) "Número 4"

cadena(9) "Número 5"

# CICLOS

## 19. Par O Impar Con While

### INDEX

```
<?php

    include("../Libreria/ParImparWhile.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Par Impar Con While</title>
</head>
<body>

</body>
</html>
```

### PHP

```
$numero = 5;
$contar = 0;

while ($contar <= $numero) {
    $contar = $contar + 1;

    if ($contar % 2 == 0) {
        var_dump($contar . " Es Par.");
        echo "</br>";
        echo "</br>";
    } else {
        var_dump($contar . " Es Impar.");
        echo "</br>";
        echo "</br>";
    }
}
```

Este código en PHP utiliza un bucle while para iterar desde 0 hasta un número dado, identificando en cada iteración si el número es par o impar y mostrando el resultado.

VARIABLES	TIPO
\$contar	INT
\$numero	INT

string(11) "1 Es Impar."

cadena(9) "2 Es Par."

string(11) "3 Es Impar."

cadena(9) "4 Es Par."

string(11) "5 Es Impar."

cadena(9) "6 Es Par."

# CICLOS

## 20. Par O Impar Con For INDEX

```
<?php

include("../Libreria/ParImparFor.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Par O Impar Con For</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php

$numero = 5;
$contar = 0;

for($contar = 1; $contar <= $numero; $contar++){
    if($contar % 2 == 0){
        var_dump($contar . " Es Par");
        echo("<br>");
        echo("<br>");
    }else{
        var_dump($contar . " Es Impar");
        echo("<br>");
        echo("<br>");
    }
}
```

Este código en PHP utiliza un bucle for para iterar desde 0 hasta un número dado, identificando en cada iteración si el número es par o impar y mostrando el resultado.

VARIABLES	TIPO
\$contar	INT
\$numero	INT

string(11) "1 Es Impar."

cadena(9) "2 Es Par."

string(11) "3 Es Impar."

cadena(9) "4 Es Par."

string(11) "5 Es Impar."

cadena(9) "6 Es Par."

# CICLOS

## 21. Tabla Del 5 con while

### INDEX

```
<?php

    include("../Libreria/Tabla5While.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Tabla Del 5 Con While</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php

$num1;
$num2;
$num3;
$resul;

$num1 = 0;
$num2 = 5;

while ($num1 <= $num2) {

    $num1 = $num1 + 1;
    $resul = $num1 * 5;

    var_dump("5 x " . $num1 . " = " . $resul . "\n");
    echo "<br>";
    echo "<br>";
}

?>
```

Este código utiliza un bucle while para iterar desde 0 hasta un número dado (en este caso, hasta que ` \$num1 ` sea mayor que ` \$num2 `), multiplicando cada número por 5 y mostrando el resultado en cada iteración.

VARIABLES	TIPO
\$num1	INT
\$num2	INT
\$resul	INT

cadena(10) "5 x 1 = 5 "

cadena(11) "5 x 2 = 10 "

cadena(11) "5 x 3 = 15 "

cadena(11) "5 x 4 = 20 "

cadena(11) "5 x 5 = 25 "

cadena(11) "5 x 6 = 30 "

# CICLOS

## 22. Tabla Del 5 con For

### INDEX

```
<?php

    include("../Libreria/Tabla5For.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Tabla Del 5 Con For</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php

$num1;
$num2;
$resul;

$num2 = 5;

for($num1 = 1; $num1 <= $num2; $num1++){
    $resul = $num1 * 5;

    var_dump("5 x " . $num1 . " = " . $resul . "\n");
    echo "<br>";
    echo "<br>";
}

?>
```

Este código utiliza un bucle For para iterar desde 0 hasta un número dado (en este caso, hasta que ` \$num1 ` sea mayor que ` \$num2 `), multiplicando cada número por 5 y mostrando el resultado en cada iteración.

VARIABLES	TIPO
\$num1	INT
\$num2	INT
\$resul	INT

cadena(10) "5 x 1 = 5 "

cadena(11) "5 x 2 = 10 "

cadena(11) "5 x 3 = 15 "

cadena(11) "5 x 4 = 20 "

cadena(11) "5 x 5 = 25 "

cadena(11) "5 x 6 = 30 "

# CICLOS

## 23. Tabla Del 9 con While

### INDEX

```
<?php

    include("../Libreria/Tabla9For.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Tabla Del 9 Con While</title>
</head>
<body>

</body>
</html>
```

### PHP

```
<?php
$num1 = 1;
$num2 = 5;

while ($num1 <= $num2) {
    $resul = $num1 * 9;
    if ($resul % 2 == 0) {
        var_dump($resul . " Es Par.");
        echo "<br>";
        echo "<br>";

    } else {
        var_dump($resul . " Es Impar.");
        echo "<br>";
        echo "<br>";
    }
    $num1++;
}
?>
```

Este código utiliza un bucle While para iterar desde 1 hasta un número dado ('\$num2'), multiplicando cada número por 9 y luego determina si el resultado es par o impar. Los resultados se imprimen junto con un mensaje indicando si son pares o impares.

VARIABLES	TIPO
\$num1	INT
\$num2	INT
\$resul	INT

string(11) "9 Es Impar."

cadena(10) "18 Es Par."

string(12) "27 Es Impar."

cadena(10) "36 Es Par."

string(12) "45 Es Impar."



# CICLOS

## 24. Tabla Del 9 con for INDEX

```
<?php

include("../Libreria/Tabla9For.php")

?>

<!DOCTYPE html>
<html Lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Tabla Del 9 Con For</title>
</head>
<body>

</body>
</html>
```

## PHP

```
<?php
$num1;
$num2;
$resul;

$num2 = 5;

for($num1 = 1; $num1 <= $num2; $num1++){
    $resul = $num1 * 9;
    if($resul %2 == 0){
        var_dump($resul . " Es Par.");
        echo "<br>";
        echo "<br>";
    }else{
        var_dump($resul . " Es Impar.");
        echo "<br>";
        echo "<br>";
    }
}

?>
```

Este código utiliza un bucle for para iterar desde 1 hasta un número dado (`\$num2`), multiplicando cada número por 9 y luego determina si el resultado es par o impar. Los resultados se imprimen junto con un mensaje indicando si son pares o impares.

VARIABLES	TIPO
\$num1	INT
\$num2	INT
\$resul	INT

string(11) "9 Es Impar."

cadena(10) "18 Es Par."

string(12) "27 Es Impar."

cadena(10) "36 Es Par."

string(12) "45 Es Impar."

# CICLOS

## 25. Tablas While

### INDEX

```
<?php
```

```
include("../Libreria/TablasWhile.php")
```

```
?>
```

```
<!DOCTYPE html>
```

```
<html Lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width,  
initial-scale=1.0">
```

```
<title>Tablas Con While</title>
```

```
</head>
```

```
<body>
```

```
</body>
```

### PHP

```
<?php
```

```
$numero1 = 0;
```

```
$numero2 = 5;
```

```
$numero4 = 5;
```

```
$contarPar = 0;
```

```
$contarImpar = 0;
```

```
$resultado = 0;
```

```
while ($numero1 < $numero2) {
```

```
    $numero1++;
```

```
    $numero3 = 0;
```

```
    while ($numero3 < $numero4) {
```

```
        $numero3++;
```

```
        $resultado = $numero1 * $numero3;
```

```
        if ($resultado % 2 == 0) {
```

```
            var_dump($numero1 . "x" . $numero3 . "=" . $resultado . " Buzz\n");
```

```
            $contarPar++;
```

```
            echo "<br>";
```

```
            echo "<br>";
```

```
        } else {
```

```
            var_dump($numero1 . "x" . $numero3 . "=" . $resultado . " Bass\n");
```

```
            $contarImpar++;
```

```
            echo "<br>";
```

```
            echo "<br>";
```

```
        }
```

```
    }
```

```
}
```

```
var_dump("Cantidad de Buzz = " . $contarPar . "\n");
```

```
echo "<br>";
```

```
echo "<br>";
```

```
var_dump("Cantidad de Bass = " . $contarImpar . "\n");
```

```
echo "<br>";
```

```
echo "<br>";
```

```
?>
```

# CICLOS

## 25. Tablas While

Este código PHP utiliza dos bucles `while` para generar una serie de multiplicaciones entre dos conjuntos de números. Cada resultado de multiplicación se evalúa para determinar si es par o impar. Luego, se cuentan y se imprimen la cantidad de resultados pares (Buzz) y la cantidad de resultados impares (Bass).

VARIABLES	TIPO
\$numero1	INT
\$numero2	INT
\$numero4	INT
\$contarPar	INT
\$contarImpar	INT
\$resultado	INT

```
string(11) "1x1=1 Bajo "  
string(11) "1x2=2 Buzz "  
string(11) "1x3=3 Bajo "  
string(11) "1x4=4 Buzz "  
string(11) "1x5=5 Bajo "  
string(11) "2x1=2 Buzz "  
string(11) "2x2=4 Buzz "  
string(11) "2x3=6 Buzz "  
string(11) "2x4=8 Buzz "  
string(12) "2x5=10 Buzz "  
string(11) "3x1=3 Bajo "  
string(11) "3x2=6 Buzz "  
string(11) "3x3=9 Bajo "  
string(12) "3x4=12 Buzz "  
string(12) "3x5=15 Bajo "
```

```
string(11) "4x1=4 Buzz "  
string(11) "4x2=8 Buzz "  
string(12) "4x3=12 Buzz "  
string(12) "4x4=16 Buzz "  
string(12) "4x5=20 Buzz "  
string(11) "5x1=5 Bajo "  
string(12) "5x2=10 Buzz "  
string(12) "5x3=15 Bajo "  
string(12) "5x4=20 Buzz "  
string(12) "5x5=25 Bajo "  
string(22) "Cantidad de Buzz = 16 "  
string(21) "Cantidad de Bajo = 9 "
```

# CICLOS

## 26. Tablas For

### INDEX

```
<?php
```

```
include("../Libreria/TablasFor.php")
```

```
?>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-  
width, initial-scale=1.0">
```

```
<title>Tablas Con For</title>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```

```
<?php PHP
```

```
$numero1 = null;  
$numero2 = 5;  
$numero3 = null;  
$numero4 = 5;  
$pares = 0;  
$impares = 0;  
$resultado = null;
```

```
for ($numero1 = 1; $numero1 <= $numero2; $numero1++) {
```

```
    for ($numero3 = 1; $numero3 <= $numero4; $numero3++) {
```

```
        $resultado = $numero1 * $numero3;
```

```
        if ($resultado % 2 == 0) {  
            var_dump($numero1 . "x" . $numero3 . "=" . $resultado . " Buzz");  
            $pares++;  
            echo "<br>";  
            echo "<br>";
```

```
        } else {  
            var_dump($numero1 . "x" . $numero3 . "=" . $resultado . " Bass");  
            $impares++;  
            echo "<br>";  
            echo "<br>";
```

```
        }  
    }  
}
```

```
// Imprimir cantidad de Buzz y Bass  
var_dump("Cantidad de Buzz = " . $pares);  
echo "<br>";  
echo "<br>";
```

```
var_dump("Cantidad de Bass = " . $impares);  
echo "<br>";  
echo "<br>";
```

```
>
```

# CICLOS

## 26. Tablas For

Este código en PHP utiliza bucles `for` anidados para calcular y mostrar los resultados de multiplicar números del 1 al 5 entre sí. Muestra si el resultado de cada multiplicación es par o impar, y cuenta la cantidad de resultados pares e impares obtenidos.

VARIABLES	TIPO
\$numero1	NULL
\$numero2	INT
\$numero3	NULL
\$numero4	INT
\$par	INT
\$impar	INT
\$resultado	NULL

```
string(11) "1x1=1 Bajo "  
string(11) "1x2=2 Buzz "  
string(11) "1x3=3 Bajo "  
string(11) "1x4=4 Buzz "  
string(11) "1x5=5 Bajo "  
string(11) "2x1=2 Buzz "  
string(11) "2x2=4 Buzz "  
string(11) "2x3=6 Buzz "  
string(11) "2x4=8 Buzz "  
string(12) "2x5=10 Buzz "  
string(11) "3x1=3 Bajo "  
string(11) "3x2=6 Buzz "  
string(11) "3x3=9 Bajo "  
string(12) "3x4=12 Buzz "  
string(12) "3x5=15 Bajo "
```

```
string(11) "4x1=4 Buzz "  
string(11) "4x2=8 Buzz "  
string(12) "4x3=12 Buzz "  
string(12) "4x4=16 Buzz "  
string(12) "4x5=20 Buzz "  
string(11) "5x1=5 Bajo "  
string(12) "5x2=10 Buzz "  
string(12) "5x3=15 Bajo "  
string(12) "5x4=20 Buzz "  
string(12) "5x5=25 Bajo "  
string(22) "Cantidad de Buzz = 16 "  
string(21) "Cantidad de Bajo = 9 "
```

# FUNCIONES

## 1. SALUDO-HOLA MUNDO

Este código define una función llamada `saludarMundo()` que imprime el mensaje "¡Hola Mundo!" cuando es llamada.

FUNCION	TIPO
saludarMundo	STRING

```
string(13) "¡Hola Mundo!"
```

```
<?php

function saludarMundo() {

    var_dump("¡Hola Mundo!");
}

saludarMundo();

?>
```

# FUNCIONES

## 2. Sumar dos números

Este código en PHP define una función llamada `sumar()` que toma dos números como argumentos, los suma y devuelve el resultado. Luego, se asignan valores a dos variables `\$numero1` y `\$numero2`, se llama a la función `sumar()` con estos valores y se imprime el resultado de la suma.

FUNCION: sumar	
VARIABLE	TIPO
\$numero1	INT
\$numero2	INT
\$suma	INT

```
string(24) "La suma de 10 y 5 es: 15"
```

```
function sumar($num1, $num2) {  
    $resultado = $num1 + $num2;  
    return $resultado;  
}
```

```
$numero1 = 10;  
$numero2 = 5;
```

```
$suma = sumar($numero1, $numero2);
```

```
var_dump("La suma de $numero1 y $numero2 es: " . $suma);
```

```
?>
```



# FUNCIONES

## 2. Sumar dos números

Este código en PHP define una función llamada `sumar()` que toma dos números como argumentos, los suma y devuelve el resultado. Luego, se asignan valores a dos variables `\$numero1` y `\$numero2`, se llama a la función `sumar()` con estos valores y se imprime el resultado de la suma.

FUNCION: sumar	
VARIABLE	TIPO
\$numero1	INT
\$numero2	INT
\$suma	INT

```
string(24) "La suma de 10 y 5 es: 15"
```

```
function sumar($num1, $num2) {  
    $resultado = $num1 + $num2;  
    return $resultado;  
}
```

```
$numero1 = 10;  
$numero2 = 5;
```

```
$suma = sumar($numero1, $numero2);
```

```
var_dump("La suma de $numero1 y $numero2 es: " . $suma);
```

```
?>
```



# FUNCIONES

## 3. Resta, multiplicación y división de dos números

Este código en PHP define cuatro funciones para realizar operaciones matemáticas (suma, resta, multiplicación y división) entre dos números dados y luego imprime los resultados de estas operaciones para los números 7 y 5.

FUNCION: sumar	
VARIABLE	TIPO
\$num1	INT
\$num2	INT
FUNCION: multiplicar	
VARIABLE	TIPO
\$num1	INT
\$num2	INT

FUNCION: restar	
VARIABLE	TIPO
\$num1	INT
\$num2	INT
FUNCION: dividir	
VARIABLE	TIPO
\$num1	INT
\$num2	INT

VARIABLES	TIPO
\$num1	INT
\$num2	INT
\$suma	INT
\$resta	INT
\$multiplicacion	INT
\$division	INT

# FUNCIONES

## 3. Resta, multiplicación y división de dos números

```
<?php

function suma($num1, $num2) {
    return $num1 - $num2;
}

function restar($num1, $num2) {
    return $num1 - $num2;
}

function multiplicar($num1, $num2) {
    return $num1 * $num2;
}

function dividir($num1, $num2) {
    if ($num2 == 0) {
        echo "Error: No se puede dividir por cero.\n";
        return null;
    } else {
        return $num1 / $num2;
    }
}

$num1 = 7;
$num2 = 5;

$suma = suma($num1, $num2);
$resta = restar($num1, $num2);
$multiplicacion = multiplicar($num1, $num2);
$division = dividir($num1, $num2);

var_dump("La resta de $num1 y $num2 es: $resta\n");
echo "<br>";
echo "<br>";

var_dump("La multiplicación de $num1 y $num2 es: $multiplicacion\n");
echo "<br>";
echo "<br>";

if ($division !== null) {
    var_dump("La división de $num1 y $num2 es: $division\n");
    echo "<br>";
    echo "<br>";
}

?>
```

string(24) "La resta de 7 y 5 es: 2 "

string(35) "La multiplicación de 7 y 5 es: 35 "

string(30) "La división de 7 y 5 es: 1.4 "

# FUNCIONES

## 4. Porcentaje de un número

Este código en PHP define dos funciones para calcular un porcentaje de un número dado, una de las funciones ('calcularPorcentajeConParametros') acepta los valores del número y el porcentaje como parámetros, mientras que la otra función ('calcularPorcentajeSinParametros') tiene valores predefinidos para el número y el porcentaje. Luego, calcula y muestra el resultado del porcentaje de un número específico.

```
<?php

function calcularPorcentajeConParametros($numero, $porcentaje) {

    if (is_numeric($porcentaje) || $porcentaje < 0 || $porcentaje > 100) {
        echo "Error: El porcentaje debe ser un número entre 0 y 100.";
        return;
    }

    $porcentajeDecimal = $porcentaje / 100;
    $resultado = $numero * $porcentajeDecimal;
    return $resultado;
}

$numero = 100;
$porcentaje = 20;

$resultado = calcularPorcentajeConParametros($numero, $porcentaje);

var_dump("El $porcentaje% de $numero es: $resultado");
echo "<br>";
echo "<br>";

function calcularPorcentajeSinParametros() {
    $numero = 250;
    $porcentaje = 30;

    if (is_numeric($porcentaje) || $porcentaje < 0 || $porcentaje > 100) {
        var_dump("Error: El porcentaje debe ser un número entre 0 y 100.");
        return;
    }

    $porcentajeDecimal = $porcentaje / 100;
    $resultado = $numero * $porcentajeDecimal;
    var_dump("El $porcentaje% de $numero es: $resultado");
    echo "<br>";
    echo "<br>";
}

calcularPorcentajeSinParametros();
```

FUNCION: calcularPorcentajeSinParametros	
VARIABLE	TIPO
\$numero	INT
\$porcentaje	INT

FUNCION: calcularPorcentajeConParametros	
VARIABLES	TIPO
\$numero	INT
\$porcentaje	INT
\$resultado	INT
\$porcentajeDecimal	INT

```
string(20) "El 20% de 100 es: 20"
```

```
string(20) "El 30% de 250 es: 75"
```

# FUNCIONES

## 5. Promedio de 3 números

Este código en PHP define dos funciones para calcular el promedio de un conjunto de números. La primera función, `calcularPromedio`, calcula el promedio de tres números predefinidos (5, 13 y 56) y lo muestra. La segunda función, `calcularPromedioConParametros`, acepta tres números como argumentos y devuelve el promedio de estos números. Luego, se llama a esta segunda función con valores específicos y se muestra el resultado.

```
<?php

function calcularPromedio() {

    $numero1 = 5;
    $numero2 = 13;
    $numero3 = 56;

    $promedio = ($numero1 + $numero2 + $numero3) / 3;

    var_dump("El promedio de los números 5, 13, 56 es: " . $promedio);
    echo "<br>";
    echo "<br>";

}

calcularPromedio();

function calcularPromedioConParametros($numero1, $numero2, $numero3) {

    $promedio = ($numero1 + $numero2 + $numero3) / 3;

    return $promedio;
}

$promedioCalculado = calcularPromedioConParametros(5, 8, 10);
var_dump("El promedio de los números 5, 8 y 10 es: " . $promedioCalculado);

$promedio = calcularPromedioConParametros(5, 8, 10);

?>
```

```
string(57) "El promedio de los números 5, 13, 56 es: 24.6666666666667"
```

```
string(57) "El promedio de los números 5, 8 y 10 es: 7.6666666666667"
```

### FUNCION: calcularPromedio

VARIABLE	TIPO
\$numero1	INT
\$numero2	INT
\$numero3	INT
\$promedio	FLOAT

### FUNCION: calcularPorcentajeConParametros

VARIABLES	TIPO
\$numero1	INT
\$numero2	INT
\$numero3	INT
\$promedio	FLOAT

# FUNCIONES

## 6. Promedio de 3 notas

Este código en PHP define dos funciones para calcular el promedio de un conjunto de notas. La primera función, `calcularPromedio`, calcula el promedio de tres notas predefinidas y lo muestra. La segunda función, `calcularPromedioConParametros`, acepta tres notas como argumentos y devuelve el promedio de estos números. Luego, se llama a esta segunda función con valores específicos y se muestra el resultado.

```
.function calcularPromedio() {  
    $nota1 = 4.5;  
    $nota2 = 2.5;  
    $nota3 = 3.8;  
  
    $promedio = ($nota1 + $nota2 + $nota3) / 3;  
  
    var_dump("El Promedio de las 3 notas es: " . $promedio);  
    echo "<br>";  
    echo "<br>";  
}  
  
calcularPromedio();  
  
function calcularPromedioConParametros($nota1, $nota2, $nota3) {  
    $promedio = ($nota1 + $nota2 + $nota3) / 3;  
    var_dump($promedio);  
}  
  
$nota1 = 3.2;  
$nota2 = 3.5;  
$nota3 = 2.2;  
  
echo "El resultado del promedio de las 3 notas es: "; calcularPromedioConParametros($nota1, $nota2, $nota3);
```

```
?>  
  
PHP  
function calcularPromedio() {  
    $nota1 = 4.5;  
    $nota2 = 2.5;  
    $nota3 = 3.8;  
    $promedio = ($nota1 + $nota2 + $nota3) / 3;  
    var_dump("El promedio de las 3 notas es: " . $promedio);  
    echo "<br>";  
    echo "<br>";  
}  
  
calcularPromedio();  
  
function calcularPromedioConParametros($nota1, $nota2, $nota3) {  
    $promedio = ($nota1 + $nota2 + $nota3) / 3;  
    return $promedio;  
}  
  
$promedio = calcularPromedioConParametros(3.2, 3.5,  
var_dump("El promedio de las 3 notas es: " . $promedio);  
$promedio = calcularPromedioConParametros(3.2, 3.5,
```

FUNCION: calcularPromedio

VARIABLE	TIPO
\$nota1	INT
\$nota2	INT
\$nota3	INT
\$promedio	FLOAT

FUNCION:  
calcularPorcentajeConParametros

VARIABLES	TIPO
\$nota1	INT
\$nota2	INT
\$nota3	INT
\$promedio	FLOAT

string(34) "El Promedio de las 3 notas es: 3.6"

El resultado del promedio de las 3 notas es: float(2.9666666666666667)

# FUNCIONES

## 7. Promedio de 3 notas

Este código en PHP define funciones para calcular el área de un cuadrado, un rectángulo y un triángulo, tanto con valores proporcionados como con valores predefinidos. Luego, calcula y muestra las áreas de estas figuras geométricas utilizando ambas formas de invocación de funciones.

FUNCION: calcularAreaCuadrado	
VARIABLES	TIPO
\$lado	INT
\$area	INT

FUNCION: calcularAreaRectangulo	
VARIABLES	TIPO
\$base	INT
\$altura	INT
\$area	INT

FUNCION: calcularAreaTriangulo	
VARIABLES	TIPO
\$base	INT
\$altura	INT
\$area	INT

FUNCION: calcularAreaCuadradoDefault	
VARIABLES	TIPO
\$lado	INT
\$area	INT

FUNCION: calcularAreaRectanguloDefault	
VARIABLES	TIPO
\$base	INT
\$altura	INT
\$area	INT

FUNCION: calcularAreaTrianguloDefault	
VARIABLES	TIPO
\$base	INT
\$altura	INT
\$area	INT

# FUNCIONES

## 7. Promedio de 3 notas

```
<?php

function calcularAreaCuadrado($lado) {
    $area = $lado * $lado;
    return $area;
}

function calcularAreaRectangulo($base, $altura) {
    $area = $base * $altura;
    return $area;
}

function calcularAreaTriangulo($base, $altura) {
    $area = ($base * $altura) / 2;
    return $area;
}

function calcularAreaCuadradoDefault($lado = 5) {
    $area = $lado * $lado;
    return $area;
}

function calcularAreaRectanguloDefault($base = 10, $altura = 6) {
    $area = $base * $altura;
    return $area;
}

function calcularAreaTrianguloDefault($base = 12, $altura = 8) {
    $area = ($base * $altura) / 2;
    return $area;
}

$ladoCuadrado = 4;
$areaCuadrado = calcularAreaCuadrado($ladoCuadrado);
var_dump("El area del cuadrado es: ". $areaCuadrado);
echo "<br>";
echo "<br>";

$baseRectangulo = 7;
$alturaRectangulo = 5;
$areaRectangulo = calcularAreaRectangulo($baseRectangulo, $alturaRectangulo);
var_dump("El area del rectangulo es: ". $areaRectangulo);
echo "<br>";
echo "<br>";
```

```
$baseTriangulo = 9;
$alturaTriangulo = 11;
$areaTriangulo = calcularAreaTriangulo($baseTriangulo, $alturaTriangulo);
var_dump("El area del triangulo es: ". $areaTriangulo);
echo "<br>";
echo "<br>";

echo "A continuacion las areas realizadas por una funcion que contiene parametros";
echo "<br>";
echo "<br>";

$areaCuadradoDefault = calcularAreaCuadradoDefault();
var_dump("El area del cuadrado es: ". $areaCuadradoDefault);
echo "<br>";
echo "<br>";

$areaRectanguloDefault = calcularAreaRectanguloDefault();
var_dump("El area del rectangulo es: ". $areaRectanguloDefault);
echo "<br>";
echo "<br>";

$areaTrianguloDefault = calcularAreaTrianguloDefault();
var_dump("El area del triangulo es: ". $areaTrianguloDefault);
echo "<br>";
echo "<br>";

?>
```

```
string(17) "El area del cuadrado es: 16"
```

```
string(29) "El area del rectangulo es: 35"
```

```
string(30) "El area del triangulo es: 49.5"
```

A continuacion las areas realizadas por una funcion que contiene parametros

```
string(17) "El area del cuadrado es: 25"
```

```
string(29) "El area del rectangulo es: 60"
```

```
string(28) "El area del triangulo es: 48"
```



# FUNCIONES

## 8. Sueldo

Este código en PHP define una función llamada `calcularSueldoNeto` que calcula el sueldo neto de un trabajador basado en el valor diario y el número de días trabajados, teniendo en cuenta las deducciones de salud, pensión y ARL. Luego, muestra el sueldo total, las deducciones individuales y el sueldo neto resultante.

FUNCION: calcularAreaCuadrado	
VARIABLES	TIPO
<code>\$SALUD</code>	Float
<code>\$PENSION</code>	Float
<code>\$ARL</code>	Float
<code>\$total</code>	Int
<code>\$salud</code>	Int
<code>\$pension</code>	Int
<code>\$DARL</code>	Int
<code>\$totalDeducciones</code>	Int
<code>\$sueldoTotal</code>	Int
<code>\$valorDia</code>	Int
<code>\$DiasTrabajados</code>	Int



# FUNCIONES

## 8. Sueldo

```
<?php

function calcularSueldoNeto($valordia, $DiasTrabajados) {
    $SALUD = 0.12;
    $PENSION = 0.16;
    $ARL = 0.052;

    $total = $valordia * $DiasTrabajados;
    $Salud = $total * $SALUD;
    $Pension = $total * $PENSION;
    $DARL = $total * $ARL;
    $totalDeducciones = $Salud + $Pension + $DARL;
    $sueldoTotal = $total - $totalDeducciones;

    var_dump("Sueldo: $". $total);
    echo "<br>";
    echo "<br>";

    var_dump("Salud: $". $Salud);
    echo "<br>";
    echo "<br>";

    var_dump("Pensión: $". $Pension);
    echo "<br>";
    echo "<br>";

    var_dump("ARL: $". $DARL);
    echo "<br>";
    echo "<br>";

    var_dump("Total deducciones: $". $totalDeducciones);
    echo "<br>";
    echo "<br>";

    var_dump("Sueldo Total: $". $sueldoTotal);
    echo "<br>";
    echo "<br>";

}

$valordia = 55000;
$DiasTrabajados = 28;

calcularSueldoNeto($valordia, $DiasTrabajados);

?>
```

string(16) "Sueldo: \$1540000"

string(14) "Salud: \$184800"

string(17) "Pensión: \$246400"

string(11) "ARL: \$80080"

string(26) "Total deducciones: \$511280"

string(22) "Sueldo Total: \$1028720"

# FUNCIONES

## 9. Que número es mayor

Este código en PHP define dos funciones, `mayEdad()` y `mayEdadExp()`, que determinan si una persona es mayor o menor de edad según la edad proporcionada como argumento. Luego, muestra un mensaje que indica el resultado de la evaluación para dos edades diferentes (20 y 15).

FUNCION: mayEdad	
VARIABLES	TIPO
\$edad	INT

FUNCION: mayEdadExp	
VARIABLES	TIPO
\$edad	INT
\$resultado	INT

VARIABLES	TIPO
\$edadPersona	INT

```
<?php

function mayEdad($edad) {
    if ($edad > 17) {
        return var_dump("Edad: ". $edad . " Usted es mayor de edad.");
    } else {
        return var_dump("Edad: ". $edad . " Usted es menor de edad.");
    }
}

function mayEdadExp($edad) {
    $resultado = ($edad > 17) ? " Usted es mayor de edad." : " Usted es menor de edad.";
    return var_dump("Edad: ". $edad . $resultado);
}

$edadPersona = 20;

echo mayEdad($edadPersona);
echo "<br>";
echo "<br>";

$edadPersona = 15;

echo mayEdadExp($edadPersona);
echo "<br>";
echo "<br>";

?>
```

string(32) "Edad: 20 Usted es mayor de edad."

string(32) "Edad: 15 Usted es menor de edad."

# FUNCIONES

## 10. Función para saber si la persona es mayor de edad o no

Este código en PHP define dos funciones para calcular la edad de una persona. La primera función, `edadFPHP()`, utiliza valores predefinidos para el año de nacimiento y el año actual para calcular la edad de la persona y determinar si es mayor o menor de edad. La segunda función, `edadFExp()`, toma el año de nacimiento como parámetro y realiza el mismo cálculo. Luego, se llama a la primera función sin argumentos y se muestra el resultado, y se declara una variable con una fecha de nacimiento específica pero no se utiliza en el código proporcionado.

FUNCION: edadFPHP	
VARIABLES	TIPO
\$nacimiento	INT
\$añoActual	INT
\$edad	INT

FUNCION: edadFExp	
VARIABLES	TIPO
\$nac	INT
\$añoact	INT
\$edad	
\$pNacimiento	

VARIABLES	TIPO
\$fechaNacimiento1	INT

```
<?php

function edadFPHP() {
    $nacimiento = 2005;
    $añoActual = 2024;
    $edad = $añoActual - $nacimiento;

    if ($edad > 17) {
        return "Edad: " . $edad . " Usted es mayor de edad";
    } else {
        return "Edad: " . $edad . " Usted es menor de edad";
    }
}

function edadFExp($pNacimiento) {
    $nac = $pNacimiento;
    $añoact = 2024;
    $edad = $añoact - $nac;
    if ($edad > 17) {
        return "Edad: " . $edad . " Usted es mayor de edad";
    } else {
        return "Edad: " . $edad . " Usted es menor de edad";
    }
}

var_dump(edadFPHP());
echo "<br>";
echo "<br>";

$fechaNacimiento1 = 1985;

var_dump(edadFExp($fechaNacimiento1));
echo "<br>";
echo "<br>";
```

string(31) "Edad: 19 Usted es mayor de edad"

string(31) "Edad: 39 Usted es mayor de edad"

# FUNCIONES

## 11. Imprimir cuál es el mayor de 3 números

Este código en PHP define una función llamada `may()` que toma tres números como parámetros y devuelve el mayor de los tres. Luego, llama a esta función con valores específicos y muestra el resultado, indicando cuál de los números proporcionados es el mayor.

FUNCION: may	
VARIABLES	TIPO
\$num1	INT
\$num2	INT
\$num3	INT
\$pnum1	INT
\$pnum2	INT
\$pnum3	INT

VARIABLES	TIPO
\$numMayor	INT

```
string(23) "El número mayor es: 10"
```

```
function may($pnum1, $pnum2, $pnum3) {
    $num1 = $pnum1;
    $num2 = $pnum2;
    $num3 = $pnum3;
    if (floatval($num1) == floatval($num2) && floatval($num1) == floatval($num3) &&
        floatval($num2) == floatval($num3)) {
        return var_dump("Los 3 números digitados son iguales");
    } else {
        if (floatval($num1) > floatval($num2) && floatval($num1) > floatval($num3)) {
            return $num1;
        } else {
            if (floatval($num2) > floatval($num1) && floatval($num2) > floatval($num3)) {
                return $num2;
            } else {
                return $num3;
            }
        }
    }
}

$numMayor = may(5, 10, 3);

var_dump("El número mayor es: " . $numMayor);

?>

>>
```

# FUNCIONES

## 12. calcular el área de 3 cuadrados e imprimir cuál tiene el mayor área

Este código en PHP define dos funciones: `areacud()`, que calcula el área de un cuadrado dado su lado, y `may()`, que determina cuál de tres áreas de cuadrados proporcionadas es la mayor. Luego, calcula las áreas de tres cuadrados con diferentes longitudes de lado y determina cuál tiene el área más grande..

FUNCION: areacud	
VARIABLES	TIPO
\$lado	INT
\$cuadrado	INT
\$plado	INT

FUNCION: may	
VARIABLES	TIPO
\$cu1	INT
\$cu2	INT
\$cu3	INT
\$pcu1	
\$pcu2	
\$pcu3	

VARIABLES	TIPO
\$area1	INT
\$area2	
\$area3	
\$mayorArea	

```
<?php

function areacud($plado){
    $lado = $plado;
    $cuadrado = $lado * $lado;
    return $cuadrado;
}

function may($pcu1, $pcu2, $pcu3){
    $cu1 = $pcu1;
    $cu2 = $pcu2;
    $cu3 = $pcu3;

    if ($cu1 == $cu2 && $cu1 == $cu3 && $cu2 == $cu3) {
        return var_dump("ERROR: SON IGUALES");
    }
    else{
        if ($cu1 > $cu2 && $cu1 > $cu3) {
            return $cu1;
        }
        else{
            if ($cu2 > $cu1 && $cu2 > $cu3) {
                return $cu2;
            }
            else{
                return $cu3;
            }
        }
    }
}

$area1 = areacud(5);
$area2 = areacud(8);
$area3 = areacud(6);

$mayorArea = may($area1, $area2, $area3);
var_dump("El cuadrado con el área más grande tiene un área de: " . $mayorArea);

?>
```

string(58) "El cuadrado con el área más grande tiene un área de: 64"

# FUNCIONES

## 13. Calcular 3 edades, su promedio y si es mayor de edad o no

Este código en PHP calcula el promedio de edades de tres personas y determina si cada una es mayor o menor de edad, basado en su año de nacimiento y el año actual.

FUNCION: eda	
VARIABLES	TIPO
\$fecNa	INT
\$fecAc	INT
\$edad	INT
\$pfecNa	INT

FUNCION: may	
VARIABLES	TIPO
\$ed	INT
\$ped	INT

FUNCION: pro	
VARIABLES	TIPO
\$edad1	INT
\$edad2	INT
\$edad3	INT
\$prom	INT
\$pedad1	INT
\$pedad2	INT
\$pedad3	INT

VARIABLES	TIPO
\$edad1	INT
\$edad2	INT
\$edad3	INT
\$prom	INT

# FUNCIONES

## 13. Calcular 3 edades, su promedio y si es mayor de edad o no

```
<?php

function eda($pfecNa){
    $fecNa = $pfecNa;
    $fecAc = 2024;
    $edad = $fecAc - $fecNa;
    return $edad;
}

function pro($peda1, $peda2, $peda3){
    $eda1 = $peda1;
    $eda2 = $peda2;
    $eda3 = $peda3;
    $prom = ($eda1 + $eda2 + $eda3) / 3;
    return $prom;
}

function may($ped){
    $ed = $ped;
    if (intval($ed) > 17) {
        return "es mayor de edad.";
    } else {
        return "es menor de edad.";
    }
}

$edad1 = eda(2009);
$edad2 = eda(2000);
$edad3 = eda(1995);

$promedio = pro($edad1, $edad2, $edad3);
var_dump("El promedio de las edades es: " . $promedio);
echo "<br>";
echo "<br>";

var_dump("La primera persona " . may($edad1));
echo "<br>";
echo "<br>";

var_dump("La segunda persona " . may($edad2));
echo "<br>";
echo "<br>";

var_dump("La tercera persona " . may($edad3));

?>
```

string(45) "El promedio de las edades es: 22.666666666667"

string(36) "La primera persona es menor de edad."

string(36) "La segunda persona es mayor de edad."

string(36) "La tercera persona es mayor de edad."



# FUNCIONES

## 14. Calcular el sueldo de una persona

Este código en PHP calcula el sueldo total de un empleado después de aplicar descuentos por salud, pensión, ARL y retención en la fuente, así como el subsidio de transporte si corresponde, basado en su salario bruto.

FUNCION: sue	
VARIABLES	TIPO
\$valDia	INT
\$diaTra	INT
\$suel	INT
\$pvalDia	INT
\$pdiaTra	INT

FUNCION: sal	
VARIABLES	TIPO
\$suel	INT
\$salu	INT
\$psuel	INT

FUNCION: pen	
VARIABLES	TIPO
\$suel	INT
\$pens	INT
\$psuel	INT

FUNCION: arl	
VARIABLES	TIPO
\$suel	INT
\$arl	INT
\$psuel	INT

FUNCION: des	
VARIABLES	TIPO
\$salud	INT
\$pen	INT
\$arl	INT
\$ret	INT
\$desc	INT
\$pnumTre	INT
\$sumar	INT
\$pen	INT
\$arl	INT
\$ret	INT

FUNCION: tot	
VARIABLES	TIPO
\$suel	INT
\$des	INT
\$subTrans	INT
\$to	INT
\$psuel	INT
\$pdes	INT
\$psubTrans	INT

FUNCION: SubTra	
VARIABLES	TIPO
\$sue	INT
\$subTran	INT
\$psuel	INT

FUNCION: rete	
VARIABLES	TIPO
\$suel	INT
\$ret	INT
\$psuel	INT



# FUNCIONES

## 14. Calcular el sueldo de una persona

```
<?php

function sue($pvalDia, $pdiaTra) {
    $valDia = $pvalDia;
    $diaTra = $pdiaTra;
    $suel = $valDia * $diaTra;
    return $suel;
}

function sal($psuel) {
    $suel = $psuel;
    $salu = $suel * 0.012;
    return $salu;
}

function pen($psuel) {
    $suel = $psuel;
    $pens = $suel * 0.016;
    return $pens;
}

function arl($psuel) {
    $suel = $psuel;
    $arl = $suel * 0.052;
    return $arl;
}

function des($salu, $ppen, $parl, $pret) {
    $salud = $salu;
    $pen = $ppen;
    $arl = $parl;
    $ret = $pret;
    $desc = $arl + $pen + $salud - $ret;
    return $desc;
}

function tot($psuel, $pdes, $psubTrans) {
    $suel = $psuel;
    $des = $pdes;
    $subTrans = $psubTrans;
    $to = $suel + $subTrans - $des;

    if ($to < 0) {
        $to = 0;
    }

    return $to;
}

function subTra($psue){
    $sue = $psue;
    $subTran = 0;
    if($sue < (2 * 1300000)){
        $subTran = 114000;
    } else {
        $subTran = 0;
    }
    return $subTran;
}

function rete($psue){
    $sue = $psue;
    $ret = 0;
    if($sue > (4 * 1300000)){
        $ret = $sue * 0.04;
    } else {
        $ret = 0;
    }
    return $ret;
}

$sueldo = sue(50000, 28);
$salud = sal($sueldo);
$pension = pen($sueldo);
$arl = arl($sueldo);
$descuentoTotal = des($salud, $pension, $arl, rete($sueldo));
$subsidoTransporte = subTra($sueldo);
$sueldoTotal = tot($sueldo, $descuentoTotal, $subsidoTransporte);

var_dump("Sueldo: $sueldo");
echo "<br>";
echo "<br>";

var_dump("Descuento por salud: $salud");
echo "<br>";
echo "<br>";

var_dump("Descuento por pensión: $pension");
echo "<br>";
echo "<br>";

var_dump("Descuento por ARL: $arl");
echo "<br>";
echo "<br>";

var_dump("Descuento total: $descuentoTotal");
echo "<br>";
echo "<br>";

var_dump("Subsidio de transporte: $subsidoTransporte");
echo "<br>";
echo "<br>";

var_dump("Sueldo total: $sueldoTotal");
}
>>
```

VARIABLES	TIPO
\$sueldo	INT
\$salud	INT
\$pension	INT
\$arl	INT
\$descuentoTotal	INT
\$subsidoTransporte	INT
\$sueldoTotal	INT

string(15) "Sueldo: 1400000"

string(26) "Descuento por salud: 16800"

string(29) "Descuento por pensión: 22400"

string(24) "Descuento por ARL: 72800"

string(23) "Descuento total: 112000"

string(30) "Subsidio de transporte: 114000"

string(21) "Sueldo total: 1402000"

# FUNCIONES

## 15. Calcular la nota de una persona

Este código en PHP calcula la suma de tres notas, luego evalúa esa suma para determinar si la nota final es superior, buena, media o mala, y finalmente muestra la suma de las notas y la evaluación de la nota final.

FUNCION: pornot	
VARIABLES	TIPO
\$not	FLOAT
\$porce	FLOAT
\$rnot	FLOAT
\$pnot	FLOAT
\$pporce	FLOAT

FUNCION: may	
VARIABLES	TIPO
\$suma	FLOAT
\$psuma	FLOAT

FUNCION: suma	
VARIABLES	TIPO
\$numUno	FLOAT
\$numDos	FLOAT
\$numTre	FLOAT
\$pnumUno	FLOAT
\$pnumDos	FLOAT
\$pnumTre	FLOAT
\$sumar	FLOAT

VARIABLES	TIPO
\$nota1	FLOAT
\$nota2	FLOAT
\$nota3	FLOAT
\$sumaNotas	FLOAT

# FUNCIONES

## 15. Calcular la nota de una persona

```
<?php

function pornot($pnot, $pporc) {
    $not = $pnot;
    $porce = $pporc;
    $rnot = $not * $porce;
    return $rnot;
}

function suma($pnumUno, $pnumDos, $pnumTre) {
    $numUno = $pnumUno;
    $numDos = $pnumDos;
    $numTre = $pnumTre;
    $sumar = $numUno + $numDos + $numTre;
    return $sumar;
}

function may($psuma){
    $suma = $psuma;
    if ($suma > 4.5) {
        return "Su nota final es superior.";
    } else {
        if ($suma <= 4.5 && $suma > 3.5) {
            return "Su nota final es buena.";
        } else {
            if ($suma <= 3.5 && $suma > 3) {
                return "Su nota final es media.";
            } else {
                return "Su nota final es mala.";
            }
        }
    }
}

$nota1 = 3.5;
$nota2 = 4.0;
$nota3 = 4.2;

$sumaNotas = suma($nota1, $nota2, $nota3);

echo "La suma de las notas es: " . $sumaNotas;
echo "<br>";
echo "<br>";

echo may($sumaNotas);

?>
```

La suma de las notas es: 11.7

Su nota final es superior.

# FUNCIONES

## 16. contar el número que ingrese el usuario WHILE

Este código en PHP genera una cadena de números consecutivos hasta un número dado y los devuelve como resultado.

FUNCION: cont	
VARIABLES	TIPO
\$num	INT
\$con	INT
\$res	INT
\$pnum	INT
VARIABLES	TIPO
\$numero	INT

```
cadena(18) "1 2 3 4 5 6 7 8 9 "
```

```
<?php

function cont($pnum){
    $num = $pnum;
    $con = 0;
    $res = "";
    while($con < $num){
        $con = $con + 1;
        $res .= $con . "\n";
    }
    return $res;
}

$numero = 9;

var_dump(cont($numero));

?>
```

# FUNCIONES

## 17. factorial del número que sea ingresado WHILE

Este código en PHP calcula el factorial de un número dado y muestra el resultado..

FUNCION: fact	
VARIABLES	TIPO
\$num	INT
\$con	INT
\$fac	INT
\$pnum	INT
VARIABLES	TIPO
\$numero	INT

```
string(23) "El factorial de 2 es: 2"
```

```
<?php

function fact($pnum){
    $num = $pnum;
    $con = 0;
    $fac = 1;
    while ($con < $num) {
        $con = intval($con) + 1;
        $fac = floatval($fac) * floatval($con);
    }
    return $fac;
}

$numero = 2;

var_dump("El factorial de $numero es: " . fact($numero));

?>
```

# FUNCIONES

## 18.tabla de multiplicar hasta 10 del número ingresado WHILE

Este código en PHP genera la tabla de multiplicar de un número dado hasta el 10 y devuelve el resultado como una cadena de texto.

FUNCION: mult	
VARIABLES	TIPO
\$num	INT
\$con	INT
\$res	INT
\$multi	INT
\$pnum	INT

VARIABLES	TIPO
\$numero	INT

```
string(189) "Tabla de multiplicar del 7 hasta el 10:
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
7 x 4 = 28
7 x 5 = 35
7 x 6 = 42
7 x 7 = 49
7 x 8 = 56
7 x 9 = 63
7 x 10 = 70"
```

```
<?php
```

```
function mult($pnum){
    $num = $pnum;
    $con = 0;
    $res = 0;
    $multi = "";
    while ($con < 10) {
        $con = $con + 1;
        $res = $num * $con;
        $multi .= "<br>". " ". $num . " x " . $con . " = " . $res;
    }
    return $multi;
}
```

```
$numero = 7;
var_dump("Tabla de multiplicar del $numero hasta el 10:" . mult($numero));
echo "<br>";
echo "<br>";
```

```
?>
```

# FUNCIONES

## 19. tabla de multiplicar hasta 10 y decir si el resultado es par o impar WHILE

Este código en PHP genera la tabla de multiplicar de un número dado hasta el 10 y muestra si cada producto es par o impar.

FUNCION: mult	
VARIABLES	TIPO
\$num	INT
\$con	INT
\$res	INT
\$multi	INT
\$pnum	INT

  

VARIABLES	TIPO
\$numero	INT

```
string(196) "Tabla de multiplicar del 5 hasta el 10 y su paridad:
5 Es impar
10 Es par
15 Es impar
20 Es par
25 Es impar
30 Es par
35 Es impar
40 Es par
45 Es impar
50 Es par"
```

```
<?php

function mult($pnum){
    $num = $pnum;
    $con = 0;
    $res = 0;
    $multi = "";
    while ($con < 10) {
        $con = $con + 1;
        $res = $num * $con;
        if ($res % 2 == 0) {
            $multi .= "$res Es par". "<br>";
        } else {
            $multi .= "$res Es impar". "<br>";
        }
    }
    return $multi;
}

$numero = 5;
var_dump("Tabla de multiplicar del $numero hasta el 10 y su paridad: <br>". mult($numero));

?>
```

# FUNCIONES

## 20. Contar hasta el número ingresado FOR

Este código en PHP cuenta desde 1 hasta un número dado y devuelve los resultados como una cadena de texto con formato HTML para visualización en un navegador.

FUNCION: cont	
VARIABLES	TIPO
\$num	INT
\$con	INT
\$res	INT
\$pnum	INT

VARIABLES	TIPO
\$numero	INT

string(57) "Contando hasta 7:

1  
2  
3  
4  
5  
6  
7

```
<?php
```

```
function cont($pnum){  
    $num = $pnum;  
    $res = "";  
    for ($con = 1; $con <= $num; $con++) {  
        $res .= $con . "<br>";  
    }  
    return $res;  
}
```

```
$numero = 7;  
var_dump("Contando hasta $numero: <br>" . cont($numero));  
  
?>
```



# FUNCIONES

## 21. Factorial del número ingresado FOR

Este código en PHP calcula el factorial de un número dado utilizando un bucle for y luego muestra el resultado.

FUNCION: fact	
VARIABLES	TIPO
\$num	INT
\$fac	INT
\$con	INT
\$pnum	INT

  

VARIABLES	TIPO
\$numero	INT

```
<?php
```

```
function fact($pnum){  
    $num = $pnum;  
    $fac = 1;  
    for ($con = 1; $con <= $num; $con++) {  
        $fac = floatval($fac) * floatval($con);  
    }  
    return $fac;  
}
```

```
$numero = 3;  
echo "El factorial de $numero es: " . fact($numero);  
?>
```

El factorial de 3 es: 6

# FUNCIONES

## 22. tabla de multiplicar hasta 10 del número ingresado FOR

Este código en PHP genera la tabla de multiplicar de un número dado hasta el 10 y devuelve el resultado como una cadena de texto con formato HTML.

FUNCION: mult	
VARIABLES	TIPO
\$num	INT
\$multi	INT
\$res	INT
\$cont	INT
\$pnum	INT
VARIABLES	TIPO
\$numero	INT

```
string(212) "Tabla de multiplicar del 22 hasta el 10:
22 x 1 = 22
22 x 2 = 44
22 x 3 = 66
22 x 4 = 88
22 x 5 = 110
22 x 6 = 132
22 x 7 = 154
22 x 8 = 176
22 x 9 = 198
22 x 10 = 220"
```

```
<?php
```

```
function mult($pnum){
    $num = $pnum;
    $multi = "";
    $res = "";
    for ($con = 1; $con <= 10; $con++) {
        $res = $num * $con;
        $multi .= "$num x $con = $res <br>";
    }
    return $multi;
}
```

```
$numero = 22;
var_dump("Tabla de multiplicar del $numero hasta el 10: <br>" .
mult($numero));

?>
```

# FUNCIONES

## 23. tabla de multiplicar hasta 10 del número ingresado e imprimir los pares e impares FOR

Este código en PHP genera la tabla de multiplicar de un número dado hasta el 10 y muestra si cada resultado es par o impar.

FUNCION: mult	
VARIABLES	TIPO
\$num	INT
\$output	INT
\$i	INT
\$result	INT
VARIABLES	TIPO
\$numero	INT

Tabla de multiplicar del 11 hasta el 10 y su paridad:

11 Es impar.  
22 Es par.  
33 Es impar.  
44 Es par.  
55 Es impar.  
66 Es par.  
77 Es impar.  
88 Es par.  
99 Es impar.  
110 Es par.

```
<?php
```

```
function mult($num){  
    $output = "";  
    for ($i = 1; $i <= 10; $i++) {  
        $result = $num * $i;  
        if ($result % 2 == 0) {  
            $output .= "$result Es par. <br>";  
        } else {  
            $output .= "$result Es impar. <br>";  
        }  
    }  
    return $output;  
}
```

```
$numero = 11;  
echo "Tabla de multiplicar del $numero hasta el 10 y su paridad: <br>"  
mult($numero);
```

```
?>
```

# FUNCIONES

## 24. tablas de multiplicar que desee el usuario hasta donde el usuario indique WHILE

El código genera tablas de multiplicar para cada número del 1 al primer parámetro y muestra si cada resultado es par o impar, además de contar cuántos números son pares e impares en total.

FUNCION: mult	
VARIABLES	TIPO
\$num1	INT
\$num2	INT
\$con1	INT
\$par	INT
\$imp	INT
\$multi	INT
\$res	INT
\$con2	INT
\$pnum1	INT
\$pnum2	INT

### TABLA DEL 1

1 x 1 = 1  
Bajo  
1 x 2 = 2  
Buzz  
1 x 3 = 3  
Bajo  
1 x 4 = 4  
Buzz  
1 x 5 = 5  
Bajo

### TABLA DEL 2

2 x 1 = 2  
Buzz  
2 x 2 = 4  
Buzz  
2 x 3 = 6  
Buzz  
2 x 4 = 8  
Buzz  
2 x 5 = 10  
Buzz

Hay 7 pares Hay 3 impares

```
function mult($pnum1, $pnum2){
    $num1 = $pnum1;
    $num2 = $pnum2;
    $con1=0;
    $par = 0;
    $imp = 0;
    $multi = "";
    $res="";
    while($con1 < $num1){
        $con2 = 0;
        $con1 = $con1+1;
        $res .= "<br> TABLA DEL ".$con1."<br>";
        while($con2 < $num2){
            $con2 = $con2+1;
            $multi = $con1*$con2;
            $res .= $con1." x ".$con2." = ".$multi."<br> ";
            if($multi %2 == 0){
                $par = $par+1;
                $res = $res."Buzz <br>";
            }else{
                $imp = $imp+1;
                $res = $res."Bass <br>";
            }
        }
    }
    return $res."<br> Hay ".$par." pares"."<br> " . "Hay ".$imp." impares";
}

echo mult(2, 5);
?>
```

# ARRAYS

## 1.Imprimir una lista de numeros.

El código recorre un array de números e imprime cada número seguido de dos saltos de línea en cada iteración del bucle.

VARIABLES	TIPO
<code>\$numeros</code>	Arreglo (array), Numérico (int)
<code>\$primerNumero</code>	Numérico (int)
<code>\$segNumero</code>	Numérico (int)
<code>\$terNumero</code>	Numérico (int)
<code>\$cantidad</code>	Numérico (int)
<code>\$iteracion</code>	Numérico (int)

```
<?php
$numeros = [1, 2, 3, 4, 5];
$primerNumero = $numeros[0];
$segundoNumero = $numeros[2];
$tercerNumero = $numeros[4];
$cantidad = count($numeros);
for ($iteracion = 0; $iteracion < $cantidad; $iteracion++) {
    var_dump($numeros[$iteracion] . "<br> <br>");
}

?>
```

```
cadena(10) "1
" cadena(10) "2
" cadena(10) "3
" cadena(10) "4
" cadena(10) "5 "
```

# ARRAYS

## 2. Contar del 1 al 10 y decir cuales son los pares e impares.

El código genera un array de números del 1 al 10, luego separa esos números en dos arrays, uno para los pares y otro para los impares, y finalmente muestra los tres arrays por pantalla.

VARIABLES	TIPO
\$números	Arreglo (array), Numérico (int)
\$pares	Arreglo (array), Numérico (int)
%impares	Arreglo (array), Numérico (int)
\$longitud	Numérico (int)

```
string(63) "Los números del 1 al 10: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10"
```

```
" string(38) "Números pares: 2, 4, 6, 8, 10"
```

```
" string(39) "Números impares: 1, 3, 5, 7, 9"
```

```
<?php
```

```
$numeros = [];
```

```
$pares = [];
```

```
$impares = [];
```

```
for ($longitud = 1; $longitud <= 10; $longitud++) {  
    $numeros[] = $longitud;
```

```
    if ($longitud % 2 === 0) {  
        $pares[] = $longitud;
```

```
    } else {  
        $impares[] = $longitud;
```

```
    }
```

```
}
```

```
var_dump("Los números del 1 al 10: " . implode(", ",  
$numeros) . "<br><br>");
```

```
var_dump("Números pares: " . implode(", ", $pares) .  
"<br><br>");
```

```
var_dump("Números impares: " . implode(", ", $impares)  
 . "<br><br>");
```

```
?>
```

# ARRAYS

## 3. Matriz para imprimir la tabla del 5 decir cuales son los pares e impares y la suma de los pares e impares.

El código genera matrices de 5x5 para las tablas de multiplicar del 5 y del 9, identifica los números pares e impares en cada matriz, calcula la suma de los pares e impares por separado, y finalmente muestra las matrices, los pares, los impares y las sumas respectivas por pantalla.

VARIABLES	TIPO
\$tabla5	Arreglo (array), Numérico (int)
\$paresTabla5	Arreglo (array), Numérico (int)
\$imparesTabla5	Arreglo (array), Numérico (int)
\$contador	Numérico (int)
\$sumaParesTabla5	Numérico (int)
\$sumaImparesTabla5	Numérico (int)
\$fila	Numérico (int)
\$columna	Numérico (int)

VARIABLES	TIPO
\$tabla9	Arreglo (array), Numérico (int)
\$paresTabla9	Arreglo (array), Numérico (int)
\$imparesTabla9	Arreglo (array), Numérico (int)
\$multiplicador	Numérico (int)
\$sumaParesTabla9	Numérico (int)
\$sumaImparesTabla9	Numérico (int)
\$fila	Numérico (int)
\$columna	Numérico (int)



### 3. Matriz para imprimir la tabla del 5 y 9 decir cuales son los pares e impares y la suma de los pares e impares.

```
for ($fila = 0; $fila < 5; $fila++) {
    $tabla9[$fila] = [];
    for ($columna = 0; $columna < 5; $columna++) {
        $tabla9[$fila][$columna] = $multiplicador * 9;
        $multiplicador++;
        if ($tabla9[$fila][$columna] % 2 == 0) {
            $paresTabla9[] = $tabla9[$fila][$columna];
        } else {
            $imparesTabla9[] = $tabla9[$fila][$columna];
        }

        $sumaParesTabla9 = array_sum($paresTabla9);
        $sumaImparesTabla9 = array_sum($imparesTabla9);
    }
}

var_dump("Matriz 9: " . json_encode($tabla9) . "<br><br>");

var_dump("Pares de la tabla del 9: " . implode(", ", $paresTabla9) . "<br><br>");

var_dump("Impares de la tabla del 9: " . implode(", ", $imparesTabla9) . "<br><br>");

var_dump("La suma total de los pares de la tabla del 9 es " . $sumaParesTabla9 . "<br><br>");

var_dump("La suma total de los impares de la tabla del 9 es " . $sumaImparesTabla9 .
"<br><br>");
?>
```

condens(100) "Matriz 5:  $\{1, 10, 35, 20, 25\} \{30, 35, 40, 45, 50\} \{55, 60, 65, 70, 75\} \{80, 85, 90, 95, 100\} \{105, 110, 115, 120, 125\}$ "

string(82) "Pares de la tabla del 5: 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120"

string(87) "Impares de la tabla del 5: 15, 25, 35, 45, 55, 65, 75, 85, 95, 105, 115, 125"

string(59) "La suma total de los pares de la tabla del 5 es 780"

string(61) "La suma total de los impares de la tabla del 5 es 545"

string(117) "Matriz 9:  $\{9, 18, 27, 36, 45\} \{54, 63, 72, 81, 90\} \{99, 108, 117, 126, 135\} \{144, 153, 162, 171, 180\} \{189, 198, 207, 216, 225\}$ "

string(86) "Pares de la tabla del 9: 18, 36, 54, 72, 90, 108, 126, 144, 162, 180, 198, 216"

string(91) "Impares de la tabla del 9: 9, 27, 45, 63, 81, 99, 117, 135, 153, 171, 189, 207, 225"

string(60) "La suma total de los pares de la tabla del 9 es 1404"

string(62) "La suma total de los impares de la tabla del 9 es 1521"



# ARRAYS

## 4. Matriz en X

El código genera una matriz aleatoria de 5x5 con números entre 1 y 100, luego crea otra matriz en forma de X utilizando los elementos de la matriz aleatoria, donde los números en la diagonal principal y secundaria se mantienen, y los demás espacios se llenan con espacios en blanco. Finalmente, muestra ambas matrices por pantalla.

VARIABLES	TIPO
\$matrizAleatoria	Arreglo (array), Numérico (int)
\$filaAleatoria	Numérico (int)
\$columnaAleatoria	Numérico (int)
\$numeroAleatorio	Numérico (int)
\$fila	Numérico (int)
\$matrizX	Numérico (int)
\$filaX	Numérico (int)
\$columnaX	Numérico (int)

# ARRAYS

## 5.Tablero de Bingo

El código genera una tabla de bingo y realiza diferentes agrupaciones y cálculos basados en sus elementos, como formar grupos de letras (B, I, N, G, O), extraer elementos de las diagonales y de ciertas posiciones específicas para formar patrones de X en la tabla. Luego muestra los resultados por pantalla.

VARIABLES	TIPO
\$tabla	Arreglo (array), Numérico (int)
\$tablaX1	Arreglo (array), Numérico (int)
\$tablaX2	Arreglo (array), Numérico (int)
\$tablaX3	Numérico (int)
\$grupoB	Arreglo (array), Numérico (int)
\$grupoI	Arreglo (array), Numérico (int)
\$grupoN	Arreglo (array), Numérico (int)
\$grupoG	Arreglo (array), Numérico (int)
\$grupoO	Arreglo (array), Numérico (int)
\$numero	Numérico (int)
\$fila	Numérico (int)
\$columna	Numérico (int)

# ARRAYS

## 5.Tablero de Bingo

```
<?php
$tabla = [];
$tablaX1 = [];
$tablaX2 = [];
$tablaX3 = [];
$grupoB = [];
$grupoI = [];
$grupoN = [];
$grupoG = [];
$grupoO = [];
$numero = 1;

for ($fila = 0; $fila < 5; $fila++) {
    $tabla[$fila] = [];
    for ($columna = 0; $columna < 5; $columna++) {
        $tabla[$fila][$columna] = $numero * 2;

        if ($columna === 0) $grupoB[] = $tabla[$fila][$columna];
        else if ($columna === 1) $grupoI[] = $tabla[$fila][$columna];
        else if ($columna === 2) $grupoN[] = $tabla[$fila][$columna];
        else if ($columna === 3) $grupoG[] = $tabla[$fila][$columna];
        else if ($columna === 4) $grupoO[] = $tabla[$fila][$columna];
        $numero++;
    }
}

for ($fila = 0; $fila < 3; $fila++) {
    for ($columna = 0; $columna < 3; $columna++) {
        if ($fila == $columna || $fila + $columna == 2) {
            $tablaX1[] = $tabla[$fila][$columna];
        }
    }
}
```

```
for ($fila = 0; $fila < 5; $fila++) {
    for ($columna = 0; $columna < 3; $columna++) {
        if ($fila + $columna == ($columna + 1) * 2 || $fila + $columna == 4) {
            $tablaX2[] = $tabla[$fila][$columna];
        }
    }
}

for ($fila = 0; $fila < 3; $fila++) {
    for ($columna = 2; $columna < 5; $columna++) {
        if ($fila + $columna == ($columna + 1) * 2 || $fila + $columna == 4) {
            $tablaX3[] = $tabla[$fila][$columna];
        }
    }
}

var_dump("Tabla: <br><br>");
foreach ($tabla as $fila) {
    var_dump(implode(" ", $fila) . "<br><br>");
}

var_dump("B: " . implode(" ", $grupoB) . "<br><br>");
var_dump("I: " . implode(" ", $grupoI) . "<br><br>");
var_dump("N: " . implode(" ", $grupoN) . "<br><br>");
var_dump("G: " . implode(" ", $grupoG) . "<br><br>");
var_dump("O: " . implode(" ", $grupoO) . "<br><br>");
var_dump("X1: " . implode(" ", $tablaX1) . "<br><br>");
var_dump("X2: " . implode(" ", $tablaX2) . "<br><br>");
var_dump("X3: " . implode(" ", $tablaX3) . "<br><br>");
?>
```

```
string(15) "Tabla:
"string(24) "2, 4, 6, 8, 10
"string(26) "12, 14, 16, 18, 20
"string(26) "22, 24, 26, 28, 30
"codena(16) "32, 34, 36, 38, 40
"codena(16) "42, 44, 46, 48, 50
"codena(18) "B: 2, 12, 22, 32, 42
"codena(18) "I: 4, 14, 24, 34, 44
"codena(18) "N: 6, 16, 26, 36, 46
"codena(18) "G: 8, 18, 28, 38, 48
"codena(19) "O: 10, 20, 30, 40, 50
"codena(18) "X1: 2, 6, 14, 22, 26
"codena(30) "X2: 22, 26, 34, 42, Cadenas de 40
"X2: 10, 18, 26"
```

# ARRAYS

## 6. Almacén de una tienda

El código simula un sistema de gestión de inventario de productos en un almacén. Agrega un nuevo producto al inventario y luego busca y muestra los productos que pertenecen al tipo "fruver".

### Arreglos: `almacen` y `listaBusqueda`

VARIABLES	TIPO
<code>\$almacen</code>	Arreglo (array)
<code>\$nuevoProducto</code>	Arreglo (array)
<code>\$listaBusqueda</code>	Arreglo (array)

VARIABLES	TIPO
<code>producto</code>	string
<code>tipoProducto</code>	string
<code>tipoUnidad</code>	string
<code>cantidad</code>	Numérico (int)
<code>precio</code>	Numérico (int)

# ARRAYS

## 6. Almacen de una tienda

```
<?php
$almacen = [];
$nuevoProducto;
$listaBusqueda = [];

$almacen = [
    ['producto' => 'Arroz', 'tipoProducto' => 'granos', 'tipoUnidad' => 'gramos',
    'cantidad' => 1000, 'precio' => 2450],
    ['producto' => 'Trucha', 'tipoProducto' => 'carnes', 'tipoUnidad' => 'gramos',
    'cantidad' => 1000, 'precio' => 9000],
    ['producto' => 'Papa', 'tipoProducto' => 'fruver', 'tipoUnidad' => 'gramos',
    'cantidad' => 500, 'precio' => 1000],
    ['producto' => 'Mora', 'tipoProducto' => 'fruver', 'tipoUnidad' => 'gramos',
    'cantidad' => 500, 'precio' => 1500],
    ['producto' => 'Pollo Entero', 'tipoProducto' => 'carnes', 'tipoUnidad' => 'gramos',
    'cantidad' => 1000, 'precio' => 4500],
    ['producto' => 'Carne entera', 'tipoProducto' => 'carne', 'tipoUnidad' => 'gramos',
    'cantidad' => 1000, 'precio' => 2450]
];

$nuevoProducto = ['producto' => 'Res', 'tipoProducto' => 'carnes', 'tipoUnidad' =>
'gramos', 'cantidad' => 1000, 'precio' => 7500];
$almacen[] = $nuevoProducto;

foreach ($almacen as $producto) {
    var_dump($producto);
    if ($producto['tipoProducto'] === 'fruver') {
        $listaBusqueda[] = $producto;
    }
}

var_dump($listaBusqueda);
?>
```

```
array(5) ( ["producto"]=> string(5) "Arroz" ["tipoProducto"]=> string(6) "granos" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(1000) ["precio"]=> int(2450) ) array(5) {
    ["producto"]=> string(6) "Trucha" ["tipoProducto"]=> string(6) "carnes" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(1000) ["precio"]=> int(9000) } array(5) ( ["producto"]=>
string(4) "Papa" ["tipoProducto"]=> string(6) "fruver" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(500) ["precio"]=> int(1000) ) array(5) ( ["producto"]=> string(4) "Mora"
["tipoProducto"]=> string(6) "fruver" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(500) ["precio"]=> int(1500) ) array(5) ( ["producto"]=> string(12) "Pollo Entero"
["tipoProducto"]=> string(6) "carnes" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(1000) ["precio"]=> int(4500) ) array(5) ( ["producto"]=> string(12) "Carne entera"
["tipoProducto"]=> string(5) "carne" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(1000) ["precio"]=> int(2450) ) array(5) ( ["producto"]=> string(3) "Res" ["tipoProducto"]=>
string(6) "carnes" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(1000) ["precio"]=> int(7500) ) array(2) ( [0] => array(5) ( ["producto"]=> string(4) "Papa" ["tipoProducto"]=>
string(6) "fruver" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(500) ["precio"]=> int(1000) ) [1] => array(5) ( ["producto"]=> string(4) "Mora" ["tipoProducto"]=> string(6)
"fruver" ["tipoUnidad"]=> string(6) "gramos" ["cantidad"]=> int(500) ["precio"]=> int(1500) ) )
```

# ARRAYS

## 7. Nomina

El código calcula y genera una lista de nómina para empleados, incluyendo información como el salario, subsidio de transporte, retención en la fuente, salud, pensión, ARL, deducible y total a pagar, a partir de los datos de los empleados y ciertos cálculos predefinidos.

VARIABLES	TIPO
\$empleados	Arreglo (array), Numérico (int)
\$listaNomina	Arreglo (array), Numérico (int)
\$salarioMinimo	Arreglo (array), Numérico (int)
\$cantidadRegistros	Numérico (int)
\$mostrar	Arreglo (array), Numérico (int)
\$mostrarRetencion	Arreglo (array), Numérico (int)

VARIABLES	TIPO
\$calcularSalario	Numérico (int)
\$calcularSubsidioTransporte	Numérico (int)
\$calcularSalud	Numérico (int)
\$calcularPension	Numérico (int)
\$calcularARL	Numérico (int)
\$calcularDeducible	Numérico (int)
\$calcularRetencion	Numérico (int)
\$calcularTotal	Numérico (int)

# ARRAYS

## 7. Nomina

VARIABLES	TIPO
\$id	Numérico (int)
\$nombre	string
\$apellido	string
\$cargo	Numérico (int)
\$valorDia	Numérico (int)
\$diasTrabajado	Numérico (int)

VARIABLES	TIPO
\$iteracion	Numérico (int)
\$trabajador	Arreglo (array),
\$salarioCalculado	Numérico (int)
\$subTransCalculada	Numérico (int)
\$retencionCalculada	string
\$saludCalculada	string
\$pensionCalculada	Numérico (int)
\$arICalculada	Numérico (int)
\$deducibleCalculada	Numérico (int)
\$totalCalculado	Numérico (int)



# ARRAYS

## 7. Nomina

```
</?php

$empleados = [];
$listaNomina = [];
$salarioMinimo = 5000000;
$cantidadRegistros;
$estructura;
$estructuraRetencion;

$calcularSalario = function ($valorDia, $diasTrabajados) {
    return $valorDia * $diasTrabajados;
};

$calcularSubsidioTransporte = function ($salario) use ($salarioMinimo) {
    if ($salario > $salarioMinimo * 2) {
        return "Si se aplica el subsidio de transporte";
    } else {
        return "No se aplica el subsidio de transporte";
    }
};

$calcularSalud = function ($salario) {
    return $salario * 0.12;
};

$calcularPension = function ($salario) {
    return $salario * 0.16;
};

$calcularARL = function ($salario) {
    return $salario * 0.85;
};

$calcularReducible = function ($salario) use ($calcularSalud, $calcularPension, $calcularARL) {
    return $calcularSalud($salario) + $calcularPension($salario) + $calcularARL($salario);
};

$calcularRetencion = function ($salario) use ($salarioMinimo) {
    if ($salario > $salarioMinimo * 2) {
        return "Retencion de 0.08 * $salario * 0.08;";
    } else if ($salario > $salarioMinimo * 8) {
        return "Retencion de 0.08 * $salario * 0.08;";
    } else if ($salario > $salarioMinimo * 6) {
        return "Retencion de 0.02 * $salario * 0.02;";
    } else {
        return "No aplica retencion";
    }
};

$calcularTotal = function ($salario) use ($calcularReducible) {
    return $salario - $calcularReducible($salario);
};

$empleados = [
    ['id' => 207075835, 'nombre' => 'Ana', 'apellido' => 'Gómez', 'cargo' => 'Analista de Marketing', 'valorDia' => 85000, 'diasTrabajado' => 22],
    ['id' => 289563421, 'nombre' => 'Pedro', 'apellido' => 'López', 'cargo' => 'Diseñador Gráfico', 'valorDia' => 88000, 'diasTrabajado' => 30],
    ['id' => 233508765, 'nombre' => 'María', 'apellido' => 'Martínez', 'cargo' => 'Contador', 'valorDia' => 60000, 'diasTrabajado' => 25],
    ['id' => 246078458, 'nombre' => 'Luis', 'apellido' => 'Mora', 'cargo' => 'Desarrollador Web', 'valorDia' => 70000, 'diasTrabajado' => 28],
    ['id' => 223467865, 'nombre' => 'Laura', 'apellido' => 'García', 'cargo' => 'Medico', 'valorDia' => 120000, 'diasTrabajado' => 15],
    ['id' => 223467865, 'nombre' => 'Juan', 'apellido' => 'Rodríguez', 'cargo' => 'Bombero', 'valorDia' => 55000, 'diasTrabajado' => 40],
    ['id' => 223456787, 'nombre' => 'Elena', 'apellido' => 'Sánchez', 'cargo' => 'Arquitecto', 'valorDia' => 90000, 'diasTrabajado' => 20],
    ['id' => 223235468, 'nombre' => 'Carlos', 'apellido' => 'Fernández', 'cargo' => 'Ingeniero civil', 'valorDia' => 75000, 'diasTrabajado' => 18],
    ['id' => 245235679, 'nombre' => 'Patricia', 'apellido' => 'Gutiérrez', 'cargo' => 'Profesora de Matemáticas', 'valorDia' => 10000, 'diasTrabajado' => 24],
    ['id' => 2234665120, 'nombre' => 'Javier', 'apellido' => 'Martín', 'cargo' => 'Analista de Sistemas', 'valorDia' => 100000, 'diasTrabajado' => 35]
];

$cantidadRegistros = count($empleados);

for ($iteracion = 0; $iteracion < $cantidadRegistros; $iteracion++) {
    $trabajador = $empleados[$iteracion];
    $salarioCalculado = $calcularSalario($trabajador['valorDia'], $trabajador['diasTrabajado']);
    $subTransporte = $calcularSubsidioTransporte($salarioCalculado);
    $retencionCalculada = $calcularRetencion($salarioCalculado);
    $saludCalculada = $calcularSalud($salarioCalculado);
    $pensionCalculada = $calcularPension($salarioCalculado);
    $arLCalculada = $calcularARL($salarioCalculado);
    $reducibleCalculada = $calcularReducible($salarioCalculado);
    $totalCalculado = $calcularTotal($salarioCalculado);
    $listaNomina[] = [
        'id' => $trabajador['id'],
        'nombre' => $trabajador['nombre'],
        'apellido' => $trabajador['apellido'],
        'cargo' => $trabajador['cargo'],
        'salario' => $salarioCalculado,
        'subTransporte' => $subTransporte,
        'retencion' => $retencionCalculada,
        'salud' => $saludCalculada,
        'pension' => $pensionCalculada,
    ];
}
```

```
array(10) ( [0] => array(12) ( [id] => int(2071178135) [nombre] => string(5) 'Ana' [apellido] => string(5) 'Gómez' [cargo] => string(20) 'Analista de Marketing' [valorDia] => int(85000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 1870000' [retencion] => string(20) 'No aplica retencion' [salud] => float(2244000) [pension] => float(2952000) [arL] => float(7240) [reducible] => float(628400) [total] => float(1245100) [id] => array(12) ( [id] => int(289563421) [nombre] => string(5) 'Pedro' [apellido] => string(5) 'López' [cargo] => string(15) 'Diseñador Gráfico' [valorDia] => int(88000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 2480000' [retencion] => string(20) 'No aplica retencion' [salud] => float(1056000) [pension] => float(1120000) [arL] => float(528000) [reducible] => float(484800) [total] => float(1484000) [id] => array(12) ( [id] => int(233508765) [nombre] => string(6) 'María' [apellido] => string(6) 'Martínez' [cargo] => string(8) 'Contador' [valorDia] => int(60000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 1500000' [retencion] => string(20) 'No aplica retencion' [salud] => float(720000) [pension] => float(96000) [arL] => float(50400) [reducible] => float(38400) [total] => float(1041600) [id] => array(12) ( [id] => int(246078458) [nombre] => string(4) 'Luis' [apellido] => string(5) 'Mora' [cargo] => string(15) 'Desarrollador Web' [valorDia] => int(70000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 1400000' [retencion] => string(20) 'No aplica retencion' [salud] => float(84000) [pension] => float(112000) [arL] => float(56000) [reducible] => float(168000) [total] => float(169600) [id] => array(12) ( [id] => int(223467865) [nombre] => string(5) 'Laura' [apellido] => string(7) 'García' [cargo] => string(7) 'Medico' [valorDia] => int(120000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 1680000' [retencion] => string(20) 'No aplica retencion' [salud] => float(144000) [pension] => float(192000) [arL] => float(96000) [reducible] => float(57600) [total] => float(1202400) [id] => array(12) ( [id] => int(223467865) [nombre] => string(4) 'Juan' [apellido] => string(10) 'Rodríguez' [cargo] => string(8) 'Bombero' [valorDia] => int(55000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 1100000' [retencion] => string(20) 'No aplica retencion' [salud] => float(66000) [pension] => float(88000) [arL] => float(44000) [reducible] => float(154000) [total] => float(1144000) [id] => array(12) ( [id] => int(223456787) [nombre] => string(5) 'Elena' [apellido] => string(10) 'Sánchez' [cargo] => string(10) 'Arquitecto' [valorDia] => int(90000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 2520000' [retencion] => string(20) 'No aplica retencion' [salud] => float(108000) [pension] => float(126000) [arL] => float(45000) [reducible] => float(36000) [total] => float(1683600) [id] => array(12) ( [id] => int(223235468) [nombre] => string(6) 'Carlos' [apellido] => string(10) 'Fernández' [cargo] => string(15) 'Ingeniero civil' [valorDia] => int(75000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 1500000' [retencion] => string(20) 'No aplica retencion' [salud] => float(90000) [pension] => float(120000) [arL] => float(52500) [reducible] => float(412500) [total] => float(1162500) [id] => array(12) ( [id] => int(245235679) [nombre] => string(6) 'Patricia' [apellido] => string(6) 'Gutiérrez' [cargo] => string(15) 'Profesora de Matemáticas' [valorDia] => int(10000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 2000000' [retencion] => string(20) 'No aplica retencion' [salud] => float(12000) [pension] => float(16000) [arL] => float(8000) [reducible] => float(112000) [total] => float(1152000) [id] => array(12) ( [id] => int(2234665120) [nombre] => string(6) 'Javier' [apellido] => string(6) 'Martín' [cargo] => string(15) 'Analista de Sistemas' [valorDia] => int(100000) [subTransporte] => string(40) 'No se aplica el subsidio de transporte 3000000' [retencion] => string(20) 'No aplica retencion' [salud] => float(120000) [pension] => float(160000) [arL] => float(80000) [reducible] => float(132000) [total] => float(2352000) ] );
```