

- Entrega 2 (1/10/25)
  - Programa 11
  - Programa 12
  - Programa 13
  - Programa 14
  - Programa 15
  - Programa 16
  - Programa 17
  - Programa 18
  - Programa 19
  - Programa 20 - En grupo (Miguel Torres & Dani Castro & Miguel Anguel Quevedo)
  - Array
  - Programa 21
  - Programa 22
  - Programa 23
  - Programa 24
  - Formularios
  - Programa 25
  - Programa 28
  - Programa 29

## Entrega 2 (1/10/25)

---

En esta entrega veremos los siguientes puntos:

- Estructura de control
  - Funciones
  - Formularios
- 

## Programa 11

---

```
<?php
$a = 10;
$b = 5;
if ($a > $b)
    echo "a es más grande que b";
else
| echo "b es más grande que a";
?>
```

Aqui hemos declarado dos variables y hemos echo que las compruebe para ver cual es mas grande y este a sido el resultado

---

a es más grande que b

Tambien se puede hacer de esta forma que tendria el mismo resultado

```
<?php
$a = 10;
$b = 5;
✓ if ($a > $b) {
    echo "a es más grande que b";
} elseif ($a == $b) {
    echo "a es igual a b";
} else {
    echo "a es más pequeño que b";
}
?>
```

## Programa 12

---

```
<?php  
  
$dia = 3; // Cambia este valor  
  
echo "Otro día<br>";  
  
switch ($dia) {  
    case 1:  
        echo "Lunes";  
        break;  
  
    case 2:  
        echo "Martes";  
        break;  
  
    case 3:  
        echo "Miércoles";  
        break;  
  
    default:  
        echo "Día no válido";  
        break;  
}  
  
?>
```

Otro día  
Miércoles

## Programa 13

```
<?php  
$age = 12;  
  
$output = match (true) {  
    $age < 2 => "Bébé",  
    $age < 13 => "Enfant",  
    $age <= 19 => "Adolescent",  
    $age >= 40 => "Adulte âgé",  
    $age > 19 => "Jeune adulte",  
};  
  
var_dump(value: $output);  
?>
```

Como podemos ver a la edad le hemos declarado un valor y despues en el array ponemos que y la edad que le pasamos es menos o igual a los valores que tengo en

el array corresponde con dada atributo del array

# Programa 14

---

```
<?php
// Programa14.php
echo "<h3>Ejemplo con for</h3>";
for ($i = 1; $i <= 10; $i++) { // de 1 a 10
    echo "Número: $i <br>";
}

echo "<h3>Ejemplo con while</h3>";
$j = 2;
while ($j <= 10) { // mientras sea menor o igual a 10
    echo "Par: $j <br>";
    $j += 2; // aumenta de 2 en 2
}

echo "<h3>Ejemplo con do...while</h3>";
$k = 5;
do {
    echo "Cuenta atrás: $k <br>";
    $k--; // resta 1
} while ($k >= 0); // hasta llegar a 0
?>
```

---

## Ejemplo con for

Número: 1

Número: 2

Número: 3

Número: 4

Número: 5

Número: 6

Número: 7

Número: 8

Número: 9

Número: 10

## Ejemplo con while

Par: 2

Par: 4

Par: 6

Par: 8

Par: 10

## Ejemplo con do...while

Cuenta atrás: 5

Cuenta atrás: 4

Cuenta atrás: 3

Cuenta atrás: 2

Cuenta atrás: 1

Cuenta atrás: 0

---

# Programa 15

```
Entregaz / Programa15.php
<?php
// Array simple de videojuegos
$videojuegos = ["Zelda", "FIFA", "Minecraft", "Call of Duty"];

// --- Foreach en su forma simple (solo valores)
echo "<h3>Lo mostramos con print_r()</h3>";
print_r(value: $videojuegos);
echo "<br>";

echo "<h3>Recorrido simple con foreach</h3><br>";
foreach ($videojuegos as $juego) {
    echo "Juego: $juego <br>";
}

// --- Foreach con clave => valor
echo "<h3>Recorrido con FOREACH clave => valor</h3>";
foreach ($videojuegos as $indice => $juego) {
    echo "Índice $indice => $juego <br>";
}

// --- Matriz bidimensional: videojuegos con género
$listajuegos = [
    ["nombre" => "Zelda", "genero" => "Aventura"],
    ["nombre" => "FIFA", "genero" => "Deportes"],
    ["nombre" => "Minecraft", "genero" => "Sandbox"],
    ["nombre" => "Call of Duty", "genero" => "Shooter"]
];

echo "<h2>Matriz bidimensional</h2>";
echo "<h3>Lo mostramos con print_r()</h3>";
print_r(value: $listajuegos);
echo "<br>";
echo "<h3>Recorrido de matriz bidimensional</h3>";
foreach ($listajuegos as $item) {
    echo "El juego {$item['nombre']} es de género {$item['genero']} <br>";
}
?>
```

#### Lo mostramos con print\_r()

Array ( [0] => Zelda [1] => FIFA [2] => Minecraft [3] => Call of Duty )

#### Recorrido simple con foreach

Juego: Zelda  
Juego: FIFA  
Juego: Minecraft  
Juego: Call of Duty

#### Recorrido con FOREACH clave => valor

índice 0 => Zelda  
índice 1 => FIFA  
índice 2 => Minecraft  
índice 3 => Call of Duty

#### Matriz bidimensional

#### Lo mostramos con print\_r()

Array ( [0] => Array ( [nombre] => Zelda [genero] => Aventura ) [1] => Array ( [nombre] => FIFA [genero] => Deportes ) [2] => Array ( [nombre] => Minecraft [genero] => Sandbox ) [3] => Array ( [nombre] => Call of Duty [genero] => Shooter ) )

#### Recorrido de matriz bidimensional

El juego Zelda es de género Aventura  
El juego FIFA es de género Deportes  
El juego Minecraft es de género Sandbox  
El juego Call of Duty es de género Shooter

# Programa 16

---

```
// Esta función calcula el precio final de un producto aplicando el IVA.  
// No recibe argumentos y devuelve el precio final con IVA redondeado a 2 decimales.  
  
<?php  
// Función simple sin argumentos  
3 references  
function precioConIVA(): float {  
    $precio = 100; // definido dentro de la función  
    $iva = 21; // definido dentro de la función (IVA del 21%)  
    $precioFinal = $precio + ($precio * $iva / 100);  
    return round(num: $precioFinal, precision: 2);  
}  
  
// Ejemplo de uso  
echo "Precio con IVA: " . precioConIVA() . " €<br>";  
?>
```

# Programa 17

---

```
php  
<?php  
2 references  
function precioConIVA (float $precio, float $iva ): float {  
    // $precio → precio base  
    // $iva → porcentaje de IVA (por defecto 21%)  
    $precioFinal = $precio + ($precio * $iva / 100);  
    return round(num: $precioFinal, precision: 2); // Redondeamos a 2 decimales  
}  
  
// Ejemplo de uso  
$precioBase = 100;  
echo "Precio base: $precioBase €<br>";  
echo "Precio con IVA (21%): " . precioConIVA(precio: $precioBase, iva: 21) . " €<br>";  
echo "Precio con IVA (10%): " . precioConIVA(precio: $precioBase, iva: 10) . " €<br>";  
?>
```

# Programa 18

---

```

<?php
// Imprime un título simple en la página
print "<br/><br/> ARGUMENTOS POR valor y por referencia <br /> ";

// Definición de la función precio_iva_referencia
// & $precio -> se pasa por referencia para modificar la variable original fuera de la función
// $iva = 0.21 -> valor por defecto del IVA (modificado según el enunciado)
1 reference
function precio_iva_referencia (&$precio /* referencia a la variable precio */, $iva = 0.21): float|int {
    // Multiplica el precio por (1 + iva) para aplicar el impuesto
    $precio *= (1 + $iva);
    // Devuelve el precio calculado (aunque la variable ya se modificó por referencia)
    return $precio;
}

// Asignamos un precio inicial
$precio = 10; // precio antes de aplicar IVA

// Mostramos el precio ANTES de llamar a la función
print "<br/><br/>1- ANTES de llamar a la función: El precio con IVA es ".$precio; // imprime 10

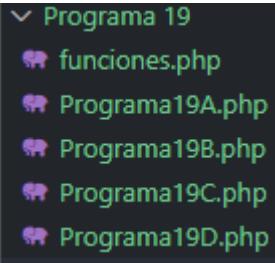
// Llamada a la función que modifica $precio por referencia usando el IVA por defecto 0.21
precio_iva_referencia(precio: &$precio);

// Mostramos el precio DESPUÉS de llamar a la función (la variable $precio ya ha cambiado)
print "<br/>2- DESPUÉS de llamar a la función: El precio con IVA es ". $precio; // imprime 12.1

// Indicamos al alumno qué return se ha usado
print "<br/><b>3- Anota en tus apuntes qué RETURN has usado, Soy Manu, y he usado el $ precio :)</b>";
?>

```

## Programa 19



## Programa 20 - En grupo (Miguel Torres & Dani Castro & Miguel Anguel Quevedo)

En este programa hemos visto el uso de varias funciones nativas de PHP que facilitan tareas comunes como trabajar con cadenas de texto, fechas, números y arrays. Gracias a ellas podemos simplificar el código y hacerlo más legible y eficiente.

- **strlen()** → devuelve el número de caracteres de una cadena.

- `strtoupper()` → convierte un string a mayúsculas.
- `strtolower()` → convierte un string a minúsculas.
- `substr()` → obtiene una subcadena desde una posición.
- `date()` → devuelve la fecha y hora en el formato especificado.
- `number_format()` → da formato numérico con decimales y separadores.
- `rand()` → genera un número aleatorio.
- `array_sum()` → suma todos los valores numéricos de un array.
- `in_array()` → comprueba si un valor está en un array.
- `implode()` → convierte un array en una cadena.

```

WES_miguelTorres > UD2 > Entrega2 > Programa20.php
1 <?php
2 echo "<h2>Programa20 - Funciones más usadas en PHP</h2><br>";
3
4 // 1. strlen() → Longitud de una cadena
5 $cadena = "Programación en PHP";
6 echo "La longitud de '$cadena' es: " . strlen(string: $cadena) . "<br>";
7
8 // 2. strtoupper() → Convertir a mayúsculas
9 echo "En mayúsculas: " . strtoupper(string: $cadena) . "<br>";
10
11 // 3. strtolower() → Convertir a minúsculas
12 echo "En minúsculas: " . strtolower(string: $cadena) . "<br>";
13
14 // 4. substr() → Extraer parte de una cadena
15 echo "Subcadena (0 a 11): " . substr(string: $cadena, offset: 0, length: 11) . "<br>";
16
17 // 5. date() → Fecha actual en formato personalizado
18 echo "Fecha actual: " . date(format: "d/m/Y H:i:s") . "<br>";
19
20 // 6. number_format() → Formatear un número con separadores
21 $precio = 12345.6789;
22 echo "Precio formateado: " . number_format(num: $precio, decimals: 2, decimal_separator: ",", thousands...".") . " €<br>";
23
24 // 7. rand() → Número aleatorio entre un rango
25 echo "Número aleatorio entre 1 y 100: " . rand(min: 1, max: 100) . "<br>";
26
27 // 8. array_sum() → Sumar todos los valores de un array
28 $numeros = [2, 4, 6, 8, 10];
29 echo "Suma de [2,4,6,8,10]: " . array_sum(array: $numeros) . "<br>";
30
31 // 9. in_array() → Verificar si un valor está en un array
32 $colores = ["rojo", "verde", "azul"];
33 echo "¿Existe 'verde' en el array de colores? " . (in_array(needle: "verde", haystack: $colores) ? "Sí" : "No") . "<br>";
34
35 // 10. implode() → Convertir array a cadena
36 echo "Colores en una cadena: " . implode(separator: ", ", array: $colores) . "<br>";
```

## Programa 20 - Funciones más usadas en PHP

La longitud de 'Programación en PHP' es: 20

En mayúsculas: PROGRAMACIÓN EN PHP

En minúsculas: programación en php

Subcadena (0 a 11): Programació

Fecha actual: 03/10/2025 14:26:12

Precio formateado: 12.345,68 €

Número aleatorio entre 1 y 100: 41

Suma de [2,4,6,8,10]: 30

¿Existe 'verde' en el array de colores? Sí

Colores en una cadena: rojo, verde, azul

## Array

---

El array es un tipo de dato que nos permite almacenar varios valores, cada miembro del array se almacena en una posición a la que se hace referencia utilizando un valor clave.

Los arrays pueden ser numéricos o asociativos

- Si en los parámetros no se indica los valores claves el array es numérico
- Si no se pasa ningún valor el array es vacío

## Programa 21

---

```

1  php
2  <?php
3
4  $array1 = array(
5      "foo" => "bar",
6      "bar" => "foo",
7  );
8
9  $array2 = [
10     "foo" => "bar",
11     "bar" => "foo",
12 ];
13 //mostrar un valor de cada uno
14 echo $array1["foo"] . "<br>";
15 echo $array2["bar"] . "<br>";
16
17 //Array multidimensional
18 $multidimensional = array(
19     "primero" => array(
20         "nombre" => "Miguel",
21         "apellido" => "Torres"
22     ),
23     "segundo" => array(
24         "nombre" => "Dani",
25         "apellido" => "Castro"
26     )
27 );
28 echo "Nombre del segundo: " . $multidimensional["segundo"]["nombre"] . "<br>";
29 echo "Apellido del primero: " . $multidimensional["primero"]["apellido"] . "<br>";
30
31 //Array creado paso a paso
32 $modulos1[0] = "Programación";
33 $modulos1[1] = "Bases de datos";
34 $modulos1[2] = "Desarrollo web en entorno servidor";
35
36 //Array creado paso a paso sin decir posición
37 $modulos2[] = "Programación";
38 $modulos2[] = "Bases de datos";
39 $modulos2[] = "Desarrollo web en entorno servidor";
40
41 echo " <br> Modulos 2: ";
42 print_r(value: $modulos2);
43
44 echo " <br> ";print_r(value: $modulos1);
45 echo " <br> ";print_r(value: $multidimensional);
46 echo " <br> ";print_r(value: $array1);print_r(value: $array2);|

```

---

```

php bar
foo
Nombre del segundo: Miguel
Apellido del primero: Castro

```

```

Modulos 2: Array ( [0] => Programación [1] => Bases de datos [2] => Desarrollo web en entorno servidor )
Array ( [0] => Programación [1] => Bases de datos [2] => Desarrollo web en entorno servidor )
Array ( [primero] => Array ( [nombre] => Miguel [apellido] => Torres ) [segundo] => Array ( [nombre] => Dani [apellido] => Castro ) )
Array ( [foo] => bar [bar] => foo ) Array ( [foo] => bar [bar] => foo )

```

# Programa 22

```
UD2 > Entrega2 > Programa22.php
1  <?php
2  $ciclos = array(
3      "DAW" => array(    "PR"    => "Programación",    "BD"    => "Bases de datos",    "DWES" => "Desarrollo web en entorno servidor"),
4      "DAM" => array(    "PR"    => "Programación",    "BD"    => "Bases de datos",    "SGE"  => "Sistemas de gestión empresarial")
5  );
6
7 //Array creado paso a paso
8 $modulos1[0] = "Programación";
9 $modulos1[1] = "Bases de datos";
10 $modulos1[2] = "Desarrollo web en entorno servidor";
11
12 // Mostrar elementos de array numérico
13 echo "<h3>Array numérico</h3>";
14 for ($i = 0; $i < count(value: $modulos1); $i++) {
15     echo "Módulo $i: " . $modulos1[$i] . "<br>";
16 }
17
18 // Recorremos el array COMPLETO
19 foreach ($ciclos as $ciclo => $modulos) {
20     echo "<h3>Ciclo: $ciclo</h3>";
21     foreach ($modulos as $codigo => $nombre) {
22         echo "[{$codigo}] => $nombre <br>";
23     }
24     echo "<br>";
25 }
26
27
```

## Array numérico

Módulo 0: Programación

Módulo 1: Bases de datos

Módulo 2: Desarrollo web en entorno servidor

## Ciclo: DAW

[PR] => Programación

[BD] => Bases de datos

[DWES] => Desarrollo web en entorno servidor

## Ciclo: DAM

[PR] => Programación

[BD] => Bases de datos

[SGE] => Sistemas de gestión empresarial

# Programa 23

```
<?php
// Definir el array con ciudades
$ciudades = array("Sevilla", "Granada", "Córdoba", "Málaga", "Cádiz");

/* Programa23: COPIA COLOCA LAS FUNCIONES DONDE CORRESPONDADN :
next($ciudades)
reset($ciudades)
prev($ciudades)
end($ciudades)
*/
// 1. reset() → primer elemento
echo "Primer elemento con : " . reset(array: &$ciudades) . "<br>";
echo "Clave actual: " . key(array: $ciudades) . "<br><br>";

// 2. next() → avanzar
echo "Siguiente Elemento con : " . next(array: &$ciudades) . "<br>";
echo "Clave actual: " . key(array: $ciudades) . "<br><br>";

// 3. prev() → retroceder
echo "Elemento con : " . prev(array: &$ciudades) . "<br>";
echo "Clave actual: " . key(array: $ciudades) . "<br><br>";

// 4. end() → último elemento
echo "Último elemento con : " . end(array: &$ciudades) . "<br>";
echo "Clave actual: " . key(array: $ciudades) . "<br><br>";

// 5. next() después del último → fuera del array
$valor = next(array: &$ciudades);
if ($valor === false && key(array: $ciudades) === null) {
    echo "El puntero está fuera del array (hemos pasado el final).<br>";
} else {
    echo "Elemento actual: $valor<br>";
}

//Vamos a forzar a forzar que se muestre
echo "Elemento con next() fuera del array: " . next(array: &$ciudades) . "<br>";
echo "Clave actual: " . key(array: $ciudades) . "<br><br>";
?>
```

Primer elemento con : Sevilla

Clave actual: 0

Siguiente Elemento con : Granada

Clave actual: 1

Elemento con : Sevilla

Clave actual: 0

Último elemento con : Cádiz

Clave actual: 4

El puntero está fuera del array (hemos pasado el final).

Elemento con next() fuera del array:

Clave actual:

## Programa 24

---

```
    echo $concesionario[$clave] . "<br>";
}

// 9. Comprobar si existe una clave
if (array_key_exists(key: "ccc3", array: $concesionario)) {
    echo "✓ Existe la clave 'c3' en el concesionario<br>";
} else{
    echo "✗ No existe la clave 'ccc3' en el concesionario<br>";
}
```

---

✓ Es un array

El concesionario tiene 3 coches

✓ El Renault Clio está en el concesionario

El Peugeot 208 tiene la clave: c4

Peugeot 208

✗ No existe la clave 'ccc3' en el concesionario

Array

```
(  
    [c2] => Seat León  
    [c3] => Renault Clio  
    [c4] => Peugeot 208  
)
```

# Formularios

---

## Programa 25

---

```
<!doctype html>
<html lang="es">
<head>
    <meta charset="utf-8">
    <title>Formulario básico</title>
</head>
<body>
    <h1>Formulario de contacto Basico Miguel Torres Villalba</h1>
    <form method="post" action="accion.php">
        <p>
            <label>Nombre:
                <input type="text" name="nombre" required>
            </label>
        </p>
        <p>
            <label>Email:
                <input type="email" name="email" required>
            </label>
        </p>
        <p>
            <label>Mensaje:<br>
                <textarea name="mensaje" rows="4" cols="40" required></textarea>
            </label>
        </p>
        <p>
            <button type="submit">Enviar</button>
        </p>
    </form>
</body>
```

```

<?php
// form.php - recoge y muestra datos del formulario

// almacenamos las variables superglobales en variables locales tras tratar los carácte
// ?? '' indica que si no se ha recibido, se pone una cadena vacía
if ($_SERVER["REQUEST_METHOD"] === "POST") {
    $nombre = htmlspecialchars(string: $_POST['nombre'] ?? '');
    $email = htmlspecialchars(string: $_POST['email'] ?? '');
    $mensaje = htmlspecialchars(string: $_POST['mensaje'] ?? '');
}
?>
<!doctype html>
<html lang="es">
<head>
    <meta charset="utf-8">
    <title>Datos recibidos</title>
</head>
<body>
    <h1>Datos recibidos del formulario</h1>
    <?php if (!empty($nombre) && !empty($email) && !empty($mensaje)): ?>
        <p><strong>Nombre:</strong> <?php echo $nombre; ?></p>
        <p><strong>Email:</strong> <?php echo $email; ?></p>
        <p><strong>Mensaje:</strong><br><?php echo nl2br(string: $mensaje); ?></p>
    <?php else: ?>
        <p style="color:red;">No se recibieron todos los datos correctamente.</p>
    <?php endif; ?>
</body>
</html>

```

## Formulario de contacto Basico Miguel Torres Villalba

Nombre:

Email:

Mensaje:

# Datos recibidos del formulario

**Nombre:** Miguel

**Email:** mtorvil0912@g.educaand.es

**Mensaje:**

sdf

## Programa 28

```
<?php
// Procesar el formulario al enviarlo
if ($_SERVER["REQUEST_METHOD"] == "POST") {

    // 1 Comprobar si el nombre está vacío
    if (empty($_POST["nombre"])) {
        echo "<p style='color:red;'>El campo nombre está vacío.</p>";
    } else {
        echo "<p>Nombre: " . htmlspecialchars(string: $_POST["nombre"]) . "</p>";
    }

    // 2 Comprobar las aficiones (checkboxes)
    if (!empty($_POST["aficiones"])) {
        $aficiones = $_POST["aficiones"];
        echo "<p>Aficiones seleccionadas: " . implode(separator: ", ", array: $aficiones) . "</p>";

        // 3 Verificar si "deporte" está entre las seleccionadas
        if (in_array(needle: "deporte", haystack: $aficiones)) {
            echo "<p>Te gusta el deporte! </p>";
        }
    } else {
        echo "<p>No has marcado ninguna afición.</p>";
    }
}
?>

<form method="post" action="">
    <label>Nombre:</label>
    <input type="text" name="nombre"
        value=<?php echo isset($_POST['nombre']) ? htmlspecialchars(string: $_POST['nombre']) : ''; ?>>
    <br><br>

    <label>Aficiones:</label><br>
    <input type="checkbox" name="aficiones[]" value="musica"
        <?php if (!empty($_POST['aficiones']) && in_array(needle: 'musica', haystack: $_POST['aficiones'])) echo "checked"; ?>>
        Música<br>

    <input type="checkbox" name="aficiones[]" value="deporte"
        <?php if (!empty($_POST['aficiones']) && in_array(needle: 'deporte', haystack: $_POST['aficiones'])) echo "checked"; ?>>
        Deporte<br>

    <input type="checkbox" name="aficiones[]" value="lectura"
        <?php if (!empty($_POST['aficiones']) && in_array(needle: 'lectura', haystack: $_POST['aficiones'])) echo "checked"; ?>>
        Lectura<br><br>

    <input type="submit" value="Enviar">
</form>
```

Nombre: Miguel

Aficiones seleccionadas: deporte

Te gusta el deporte!

Nombre:

Aficiones:

- Música
- Deporte
- Lectura

## Programa 29

```
nuregaz / Programa29 / iprocesa.php
<html>
    </head>
    <body>
<?php

    //Se muestra solo si se ha pulsado el botón ENVIAR (isset)
    if (isset($_POST['boton_enviar'])) {
        $nombre = $_POST['nombre'];
        $modulos = $_POST['modulos'];
        print "Nombre: ".$nombre."<br />";
        foreach ($modulos as $modulo) { //Warning key undefined
            print "Modulo: ".$modulo."<br />";
        }
    }
    else {
?>    <!-- ACTION: le indicamos PHP_SELF ya que se va a REDIRIGIR a este mismo archivo -->
        <form name="input"      action=<?php $_SERVER['PHP_SELF'];?>"      method="post">
            Nombre del alumno: <input type="text" name="nombre" /><br />

            <p>Módulos que cursa:</p>
            <input type="checkbox" name="modulos[]" value="DWES" />
            Desarrollo web en entorno servidor<br />
            <input type="checkbox" name="modulos[]" value="DWEC" />
            Desarrollo web en entorno cliente<br />
            <br />

            <!-- Ahora el botón submit tiene un atributo name "boton_enviar" -->
            <input type="submit" value="Enviar" name="boton_enviar"/>
        </form>
<?php
    }
?>
    </body>
</html>
```

Podemos ver que en el código ahí comentado un Warming para que veamos que si en el apartado de señalar los modulos si no señas ninguno y envias el formulario salta un Warming

---

**Warning:** Undefined array key "modulos" in **C:\xampp\htdocs\DWES\DWES\_miguelTorres\UD2\Entrega2\Programa29\1procesa.php** on line 17  
Nombre: Miguel

**Warning:** foreach() argument must be of type array|object, null given in **C:\xampp\htdocs\DWES\DWES\_miguelTorres\UD2\Entrega2\Programa29\1procesa.php** on line 19