

Laboratorio II / Laboratorio de Software II 2024

Clase Teórica 04: Funciones definidas por el usuario, Formularios



Docente: Myriam Ruiz
Licenciada en Informática
Profesora en Computación y Matemática
Programadora Universitaria



Funciones Definidas por el usuario

- En la Clase 02 vimos que existían 2 tipos de Funciones, las Predefinidas del lenguaje y las creadas por nosotros, conocidas como Definidas por el usuario.
- Sobre estas últimas tratará esta clase.
- Las Funciones Definidas por el Usuario se vuelven una necesidad cuando queremos reutilizar un código en particular más de una vez, lo que favorece también la aplicación de Divide & Conquer y un fácil testeo y mantenimiento.

Funciones Definidas por el usuario

- Sintaxis para creación / declaración:

```
function nombreFuncion ($parametro1, $parametro2, etc.)  
{  
    // cuerpo de la función  
    return $valorDevuelto;  
}
```

Importante: las buenas prácticas indican que los nombres de función se escriben en **camelCase** y la llaves **{ }** se ubican donde se observan.

Funciones Definidas por el usuario

- **Aclaración:** Consideraremos 2 tipos de function
 - **Función:** function que devuelve (return) algo asignable a una variable.
 - **Procedimiento:** function que No devuelve algo asignable, pero puede producir side-effects (efectos secundarios):
 - Ej: Escribir en un archivo, Mostrar por pantalla, Modificar una variable aún luego de salir del procedimiento.

Funciones Definidas por el usuario

- **Cómo llamar / invocar / utilizar a cada tipo de función**
- **Función:**
`$variable = nombreFuncion ($parametro1, $parametro2, etc.);`
- **Procedimiento:**
`nombreProcedimiento ($parametro1, $parametro2, etc.);`

Ejemplo de Función con Retorno

```
<?php
    include 'html/header.html';
    function areaTriangulo ($base, $altura)
    {
        $area = $base * $altura / 2;
        return $area;
    }

    $area1 = areaTriangulo(3, 4);
    echo '<p>Área1 = ' . $area1 . '</p>';
    $area2 = areaTriangulo(5, 4);
    echo '<p>Área2 = ' . $area2 . '</p>';

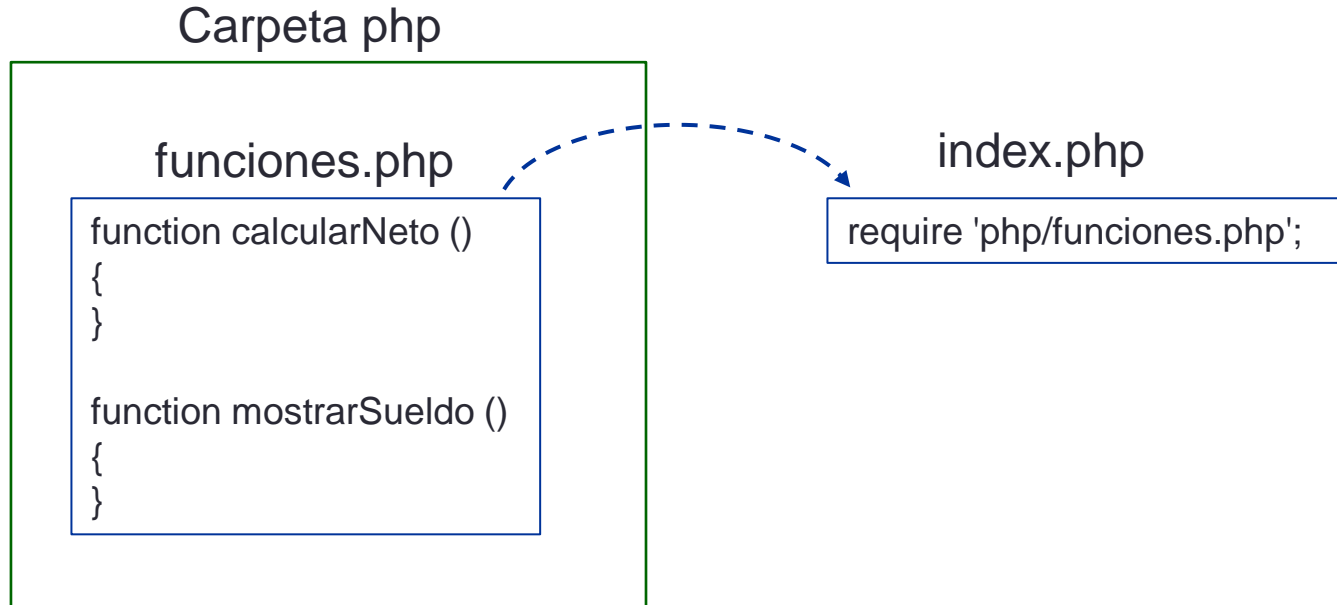
    include 'html/footer.html';
?>
```



Funciones Definidas por el usuario - Otro Ejemplo

- En un archivo llamado funciones.php, desarrollar:
 - La función **calcularNeto**, que reciba el sueldo bruto de un empleado y **devuelva** el sueldo neto (bruto menos descuentos), sabiendo que:
 - Se le descuenta 11% en concepto de aporte jubilatorio
 - Por ley 19032 se descuenta 3%
 - Y por obra social se descuenta otro 3%
 - Otra función **mostrarSueldo**, que **sólo muestre** (side-effect) el sueldo bruto y el neto.
- En otro archivo index.php, requerir el archivo funciones.php y probar con 2 sueldos brutos aleatorios diferentes

Funciones Definidas por el usuario - Ejemplo



Funciones Definidas por el usuario - Ejemplo

funciones.php

```
<?php
    const JUBILACION = 11; // visible en todo el código
    const LEY19032 = 3;
    const OS = 3;
    function calcularSueldoNeto ($sueldoB)
    {
        $desJubilacion = $sueldoB * JUBILACION/100;
        $desLey = $sueldoB * LEY19032/100;
        $desOS = $sueldoB * OS/100;
        $neto = $sueldoB - ($desJubilacion + $desLey + $desOS);
        return $neto;
    }

    function mostrar($sueldoB, $sueldoN)
    {
        echo '<p>Sueldo Bruto: $' . number_format($sueldoB, 2, ',', '.') . '</p>';
        echo '<p>Sueldo Neto: $' . number_format($sueldoN, 2, ',', '.') . '</p>';
    }
?>
```

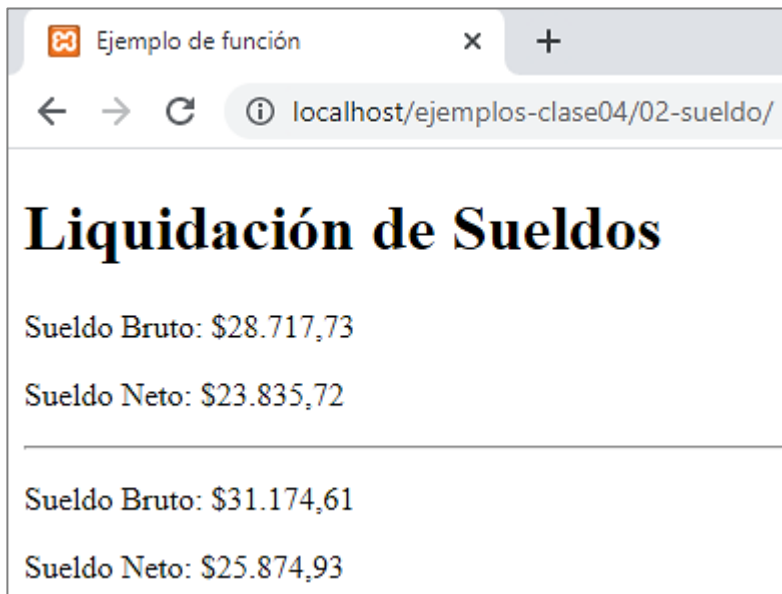
Funciones Definidas por el usuario - Ejemplo

index.php

```
<?php
    require_once 'html/header.html';
    require_once 'php/funciones.php'; // necesario para usar las funciones
    $sueldoBruto1 = mt_rand(10000*100, 60000*100)/100; // 2 decimales
    $sueldoNeto1 = calcularSueldoNeto($sueldoBruto1);
    mostrar($sueldoBruto1, $sueldoNeto1);
    echo '<hr>';
    $sueldoBruto2 = mt_rand(10000*100, 60000*100)/100; // 2 decimales
    $sueldoNeto2 = calcularSueldoNeto($sueldoBruto2);
    mostrar($sueldoBruto2, $sueldoNeto2);
    require_once 'html/footer.html';
?>
```

Funciones Definidas por el usuario - Ejemplo

index.php



The screenshot shows a web browser window with a single tab titled "Ejemplo de función". The address bar displays "localhost/ejemplos-clase04/02-sueldo/". The page content is titled "Liquidación de Sueldos" in a large, bold, black serif font. Below the title, there are two sections of data. The first section shows "Sueldo Bruto: \$28.717,73" and "Sueldo Neto: \$23.835,72". The second section, separated by a horizontal line, shows "Sueldo Bruto: \$31.174,61" and "Sueldo Neto: \$25.874,93".

Liquidación de Sueldos	
Sueldo Bruto:	\$28.717,73
Sueldo Neto:	\$23.835,72
<hr/>	
Sueldo Bruto:	\$31.174,61
Sueldo Neto:	\$25.874,93

Aclaraciones sobre Funciones

- PHP no requiere que la función esté declarada en un momento previo a su uso (puede declararse luego).
- No declarar constantes en una función:
 - Porque al llamar a la función varias veces, se intentará declarar cada vez a las constantes y eso dará error, indicando que la constante ya fue definida.

Ámbito de las Variables

- **Local:** Las variables se ven / utilizan / acceden sólo en una porción de código, por ejemplo dentro de una función.
- **Global:** Las variables se declaran fuera de una función y se ven / utilizan / acceden en cualquier lugar del código, fuera o dentro de funciones, etc. (no las usaremos)
- **Superglobales:** Las variables se pueden acceder desde todo el código.

Variables Locales

- Las variables locales no existen fuera de la porción de código en la que se creó.

```
<article>
  <?php
    function generarEdad ()
    {
      $edad = mt_rand(1,100); // $edad existe dentro de la función
    }

    generarEdad();
    echo '<p>Edad: ' . $edad . '</p>'; // en este contexto no existe $edad
  ?>
</article>
```

Notice: Undefined variable: edad in C:\xampp20\htdocs\clase04\04-variable-local\index.php on line 22

Variables Locales

- Se puede comunicar el resultado de la variable mediante return.

```
<article>
  <?php
    function generarEdad ()
    {
        $edad = mt_rand(1,100); // $edad existe dentro de la función
        return $edad;
    }

    $miEdad = generarEdad();
    echo '<p>Edad: ' . $miEdad . '</p>';
  ?>
```

← → ↻ ⓘ localhost/clase04/04-variable-local/

Función con variable local

Edad: 24

Variables Superglobales

- Las Variables **Superglobales** son variables predefinidas en PHP, que están disponibles en todos los ámbitos a lo largo del script.
- Estas variables son arreglos (arrays) y se acceden como tales:
 - `$_SERVER`
 - `$_GET`
 - `$_POST`
 - `$_FILES`
 - `$_COOKIE`
 - `$_SESSION`
 - `$_REQUEST`
 - `$_ENV`

Formularios

- En Laboratorio I se vio que cada elemento de formulario tenía atributos/propiedades, que quizá colocábamos porque se indicaba, sin pensar mucho en su utilidad.
- En Laboratorio II, la correcta escritura de estos permitirá luego poder procesar, del lado del servidor, lo que se haya enviado.

Formularios Básico - Partes

- Cada formulario presenta siempre estas partes

```
<article>
  <form action="donde" method="como">
    <label for="identificador1">Rótulo1</label>
    <input type="tipo" id="identificador1" value="Valor por Defecto1" name="nombre1">
    <label for="identificador2">Rótulo2</label>
    <input type="tipo" id="identificador2" value="Valor por Defecto2" name="nombre2">
    <input type="submit" value="Texto que dirá1">
    <input type="reset" value="Texto que dirá2">
  </form>
</article>
```

Formularios Básico - Partes

```
<form action="" method="">
```

```
</form>
```

- **action** – indica hacia donde se enviarán los datos del formulario.
- **method** – indica de que manera serán enviados.

Formularios Básico - Partes - action

```
<form action=" ../php/procesar.php" method="">
```

```
</form>
```

- En este caso el **action** indica que los datos serán enviados a la página procesar.php de nuestro sitio, ubicada en la carpeta php.

Formularios Básico - Partes - method = "get"

```
<form action=" ../php/procesar.php" method="get">
```

```
</form>
```

- En este caso el `method="get"` indica que los datos serán enviados añadiéndose a la dirección de la página indicada en `action`
- Los datos se almacenarán en el arreglo `$_GET`
- Se usa cuando no importa si el dato se puede ver o intersectar (Ej: son números que se usan en un cálculo no sensible o estamos Testeando el formulario)

Formularios Básico - Partes - method = "get"

- Enviaremos datos con method = "get", desde index.html hacia procesar.php que se encuentra en la carpeta php.

```
<article>
  <form action="php/procesar.php" method="get">
    <legend>Ingrese sus datos</legend>
    <label for="ape">Apellido:
      <input type="text" name="apel" id="ape">
    </label>
    <label for="fecha">Fecha Nacimiento:
      <input type="date" name="fecnac" id="fecha">
    </label>
    <input type="submit" value="Cargar">
    <input type="reset" value="Limpiar">
  </form>
```

clave del array \$_GET



Formularios Básico - Partes - method = "get"

The screenshot shows a web browser window with the title 'Formulario 1'. The address bar displays the URL: `localhost/clase04/06-formulario-ingreso/php/procesar.php?apel=Pérez&fecnac=2020-09-22`. The page content includes the heading 'Cargado de Cumpleaños' and the instruction 'Ingresa sus datos'. There are two input fields: 'Apellido:' with the value 'Pérez' and 'Fecha Nacimiento:' with the value '22/09/2020'. Below these fields are two buttons: 'Cargar' and 'Limpiar'. A calendar widget is open, showing the month of September 2020, with the date '22' selected. A blue arrow points from the 'Cargar' button to the query parameters in the browser's address bar.

Los name de los elementos se ven al lado de la url como variables que tienen asignados los valores seleccionados o escritos en los input

Al enviar, todos los datos se almacenan en el arreglo superglobal `$_GET`

Formularios Básico - Recepción de formulario

```
pp20 > htdocs > clase04 > 06-formulario-ingreso > php > 🐘 procesar.php
```

```
<?php
    include '../html/cabecera.html';
    include '../html/inicio-cuerpo.html';

    if (!empty($_GET['apel']) && !empty($_GET['fecnac'])) { // si no están vacío
        echo '<p>con var_dump se observa el arreglo que llega: </p>';
        var_dump($_GET);
        $apellido = $_GET['apel'];
        $fecha = $_GET['fecnac'];
        echo '<p>Apellido ingresado: ' . $apellido . '</p>';
        echo '<p>Fecha de Nacimiento: ' . $fecha . '</p>';
    } else {
        echo '<p>No introdujo Apellido o fecha de nacimiento</p>';
    }

    include '../html/pie.html';
?>
```


Formularios Básico - Recepción de formulario

← → ↻ ⓘ localhost/clase04/06-formulario-ingreso/php/procesar.php?apel=Pérez&fecnac=2020-09-22

Cargado de Cumpleaños

con var_dump se observa el arreglo que llega:

```
array(2) { ["apel"]=> string(6) "Pérez" ["fecnac"]=> string(10) "2020-09-22" }
```

Apellido ingresado: Pérez

Fecha de Nacimiento: 2020-09-22

Todos los Derechos Reservados

Contenido del Arreglo \$_GET

La fecha (y todo elemento) se recibe como string

Anexo: Por qué Pérez ocupa un string de 6?

- Esto se debe a que el carácter **é** (con tilde) es un carácter especial conocido como ***carácter multibyte***. Esto significa que, dependiendo de la codificación que estés utilizando, algunos caracteres pueden ocupar más de un byte.
- Por ejemplo, en la codificación **UTF-8**, que es la más común en la web, el carácter **é** ocupa **2 bytes** en lugar de 1, lo que hace que el total de caracteres almacenados en el string sea 6, aunque visualmente parezca que debería ser 5.

Función empty

- `empty(mixed $var): bool`
- Determina si una variable es considerada vacía.
 - Se considera vacía si no existe, es decir, si no fue creada
 - O si su valor es igual a false
- Devuelve true si está vacía y false si no lo está

Formularios Básico - Partes - method = "post"

```
<form action=" ../php/procesar.php" method="post">
```

- En este caso el `method="post"` indica que los datos serán enviados encerrados dentro de la petición HTTP, de manera no visible en la dirección del navegador.
- Se almacenarán en el arreglo `$_POST`
- Se usa para mandar información de manera privada o cuando se manejan grandes volúmenes de datos (enviar archivos de imagen por ej.)

Formularios Básico - Partes - method = "post"

- Como ejemplo, veremos un formulario que servirá para pasar datos sensibles, como ser número de tarjeta de crédito y el código de verificación de la misma. Estos datos no debieran verse al lado de la dirección de la página.
- Si se reciben correctamente enviar el mensaje "Datos completos", sino enviar "Faltan datos"



Formularios Básico - Partes - method = "post"

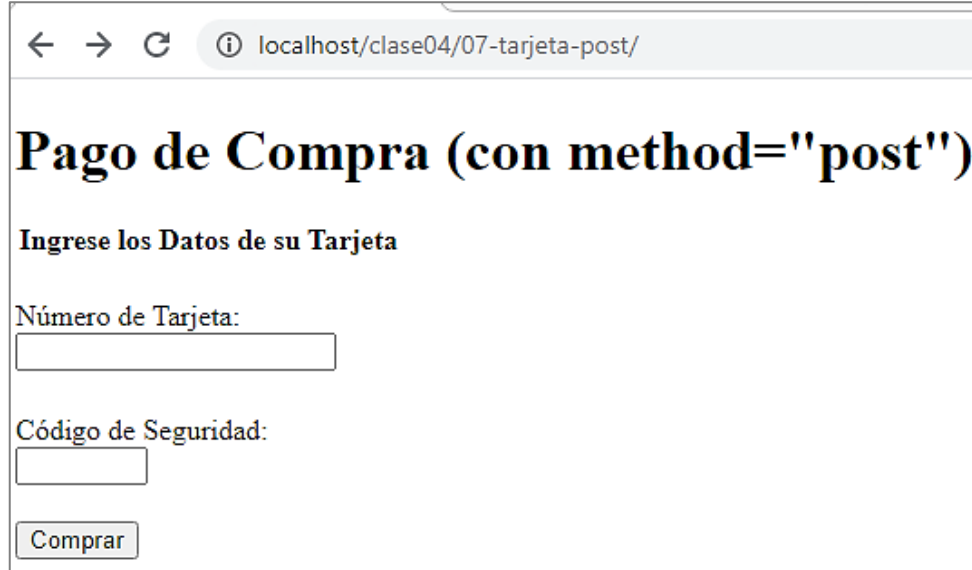
- Tarjeta

index.html

```
<article>
  <form action="php/validar.php" method="post">
    <legend>Ingresa los Datos de su Tarjeta</legend>
    <label for="num">Número de Tarjeta:</label>
    <input type="text" name="numero" id="num" autocomplete="off">
    <label for="cod">Código de Seguridad:</label>
    <input type="number" name="codigo" id="cod" min="100" max="999" autocomplete="off">
    <input type="submit" value="Comprar">
  </form>
</article>
```

Formularios Básico - Partes - method = "post"

- Tarjeta



A screenshot of a web browser window. The address bar shows 'localhost/clase04/07-tarjeta-post/'. The page title is 'Pago de Compra (con method="post")'. Below the title, the text 'Ingrese los Datos de su Tarjeta' is displayed. There are two input fields: 'Número de Tarjeta:' and 'Código de Seguridad:'. At the bottom, there is a 'Comprar' button.

← → ↻ ⓘ localhost/clase04/07-tarjeta-post/

Pago de Compra (con method="post")

Ingrese los Datos de su Tarjeta

Número de Tarjeta:

Código de Seguridad:

Formularios Básico - Partes - method = "post"

- Tarjeta

validar.php

```
<?php
    include '../html/cabecera.html';
    include '../html/inicio-cuerpo.html';

    // controlar que no estén vacíos
    if (!empty($_POST['numero']) && !empty($_POST['codigo'])) {
        echo '<p>Datos Completos</p>';
    } else {
        echo '<p>Faltan datos</p>';
    }

    include '../html/pie.html';
?>
```


Formularios Básico - Partes - method = "post"

- Tarjeta



- Observe que por haber usado method = "post", no hay variables en la url.

Formularios Accesible – Controlar

- Es muy importante controlar en el archivo php si se está recibiendo datos en el arreglo `$_GET` (o `$_POST`) antes de usar esos datos.
- Si por error un usuario entrara a una página sin tener cargado el arreglo superglobal `$_GET` o `$_POST`, verá:

```
array(0) { }
```

Notice: Undefined index: numero1 in C:\xampp19\htdocs\ejemplosClase03\formularioClase\php\procesar.php on line 5

Notice: Undefined index: numero2 in C:\xampp19\htdocs\ejemplosClase03\formularioClase\php\procesar.php on line 6

- Por eso en todos los casos se chequea al menos con `!empty`.

Formularios - Elementos

- Veremos a continuación algunos **Elementos** de formulario, para comprender cada uno de sus atributos, pero se puede extender a los demás. El resto se encuentra en el archivo Anexo Formularios.
- Todos los formularios **deben ser accesibles**, es decir, debe utilizarse la etiqueta `<label for="id"> Rótulo </Label>` donde id debe coincidir con el id del elemento asociado.

Formularios – Ejemplo con select

- En un formulario introducir nombre, edad, y seleccionar una provincia. Mandar esos datos a otra página llamada chequear.php, controlar que se estén recibiendo los datos y armar una página que muestre el contenido del array `$_POST`

Formularios – Ejemplo con select

- Página con formulario:

```
<article>
  <form action="php/chequear.php" method="post">
    <legend>Ingrese sus datos Personales</legend>
    <label for="nom">Nombre:</label>
    <input type="text" name="nombre" id="nom" placeholder="Apellido Nombre">
    <label for="ed">Edad:</label>
    <input type="number" name="edad" id="ed" min="1" max="110">
    <label for="prov">Seleccione su Provincia:</label>
    <select name="provincia" id="prov">
      <option value="Catamarca">Catamarca</option>
      <option value="Jujuy">Jujuy</option>
      <option value="Santiago">Santiago del E.</option>
      <option value="Salta">Salta</option>
      <option value="Tucuman" selected>Tucumán</option>
    </select>
    <input type="submit" value="Guardar">
  </form>
```

Formularios – Ejemplo con select

- Página con formulario:



A screenshot of a web browser window. The address bar shows 'localhost/clase04/08-select-post/'. The page title is 'Ingreso de datos personales'. Below the title is the instruction 'Ingrese sus datos Personales'. There are three input fields: 'Nombre:' with a placeholder 'Apellido Nombre', 'Edad:' with an empty box, and 'Seleccione su Provincia:' with a dropdown menu showing 'Tucumán'. At the bottom is a 'Guardar' button.

← → ↻ ⓘ localhost/clase04/08-select-post/

Ingreso de datos personales

Ingrese sus datos Personales

Nombre:

Edad:

Seleccione su Provincia:

Formularios – Ejemplo con select

- Página chequear.php (recibe el arreglo del formulario)

```
<?php
    include '../html/cabecera.html';
    include '../html/inicio-cuerpo.html';

    // controlar que los elementos no estén vacíos
    if (!empty($_POST['nombre']) && !empty($_POST['edad']) && !empty($_POST['provincia'])) {
        var_dump($_POST);
        echo '<p>Nombre: '. $_POST['nombre'] .'</p>';
        echo '<p>Edad: '. $_POST['edad'] .'</p>';
        echo '<p>Provincia: '. $_POST['provincia'] .'</p>';
    } else {
        echo '<p>Faltan datos</p>';
    }

    include '../html/pie.html';
?>
```

Formularios – Ejemplo con select

- Página chequear.php (recibe el arreglo del formulario)



← → ↻ ⓘ localhost/clase04/08-select-post/php/chequear.php

Ingresa sus datos personales

```
array(3) { ["nombre"]=> string(13) "Pérez Lorena" ["edad"]=> string(2) "20" ["provincia"]=> string(5) "Jujuy" }
```

Nombre: Pérez Lorena

Edad: 20

Provincia: Jujuy

Todos los Derechos Reservados

Formularios – Ejemplo con `input type="radio"` y `"check"`

- Crear un formulario que permita elegir tipo de prenda y talla para las mismas.
- Otra página llamada compra.php, recibirá los datos, verificará que existan y mostrará la selección realizada. Como forma de visualizar el arreglo que se recibe se usa `var_dump`, pero en un trabajo real no se usa.

Formularios – Ejemplo con input type="radio" y "check"

- El formulario html tiene input de tipo radio y checkbox

```
<article>
  <h2>Seleccione prendas</h2>
  <form action="php/compra.php" method="post">
    <fieldset>
      <legend>Tipo de Prenda</legend>
      <label for="cami">Camisa</label>
      <input type="checkbox" value="Camisa" name="cami" id="cami">
      <label for="reme">Remera</label>
      <input type="checkbox" value="Remera" name="reme" id="reme">
      <label for="saco">Saco</label>
      <input type="checkbox" value="Saco" name="saco" id="saco">
      <label for="camp">Campera</label>
      <input type="checkbox" value="Campera" name="camp" id="camp">
    </fieldset>
    <fieldset>
      <legend>Talle</legend>
```

Formularios – Ejemplo con input type="radio" y "check"

```
<legend>Talle</legend>
<label for="txs">XS</label>
|   <input type="radio" value="XS" name="talle" id="txs">
<label for="ts">S</label>
|   <input type="radio" value="S" name="talle" id="ts">
<label for="tm">M</label>
|   <input type="radio" value="M" name="talle" id="tm">
<label for="tl">L</label>
|   <input type="radio" value="L" name="talle" id="tl">
<label for="txl">XL</label>
|   <input type="radio" value="XL" name="talle" id="txl">
<label for="txxl">XXL</label>
|   <input type="radio" value="XXL" name="talle" id="txxl">
<label for="txxl">XXXL</label>
|   <input type="radio" value="XXXL" name="talle" id="txxxl">
<label for="t4xl">4XL</label>
|   <input type="radio" value="4XL" name="talle" id="t4xl">
<label for="t5xl">5XL</label>
|   <input type="radio" value="5XL" name="talle" id="t5xl">
</fieldset>
<input type="submit" value="Comprar">
```

Formularios – Ejemplo con `input type="radio"` y `"checkbox"`

← → ↻ ⓘ localhost/clase04/09-prendas-radio-check/ ☆ ⬇ ⬇ ⬇ ⬇ ⬇ ⬇ ⬇ ⬇ ⬇ ⬇

Tiendas San Luis

Seleccione prendas

Tipo de Prenda—

Camisa ☒ Remera ☒ Saco ☒ Campera ☐

Talle—

XS ☐ S ☐ M ☐ L ☐ XL ☐ XXL ☒ XXXL ☐ 4XL ☐ 5XL ☐

Comprar

Formularios – Ejemplo con `input type="radio"` y `"checkbox"`

- La página `compra.php`, controla que no estén vacíos para mostrarlos

```
<?php
// se debería incluir un encabezado
var_dump($_POST); // ver qué índices tiene, por lo general no se muestra
// controlar que los elementos no estén vacíos
if ((!empty($_POST['cami']) || !empty($_POST['reme']) || !empty($_POST['saco']) || !empty($_POST['camp']))
    && !empty($_POST['talle'])) {
    echo '<h2>Tipo de prenda:</h2>';
    if (!empty($_POST['cami'])) {
        echo '<p>' . $_POST['cami'] . '</p>';
    }
    if (!empty($_POST['reme'])) {
        echo '<p>' . $_POST['reme'] . '</p>';
    }
    if (!empty($_POST['saco'])) {
        echo '<p>' . $_POST['saco'] . '</p>';
    }
    if (!empty($_POST['camp'])) {
        echo '<p>' . $_POST['camp'] . '</p>';
    }
    echo '<h2>Talle: </h2>';
    echo '<p>' . $_POST['talle'] . '</p>';
} else {
    echo '<p>Debe seleccionar al menos una prenda y el talle</p>';
}
```

Formularios – Ejemplo con `input type="radio"` y `"checkbox"`

- Lo que muestra compra.php (el `var_dump` se usa al sólo fin de que el programador constate lo que viene mientras prueba el código)

```
← → ↻ ⓘ localhost/clase04/09-prendas-radio-check/php/compra.php  
array(4) { ["cami"]=> string(6) "Camisa" ["reme"]=> string(6) "Remera" ["saco"]=> string(4) "Saco" ["talle"]=> string(3) "XXL" }
```

Tipo de prenda:

Camisa

Remera

Saco


Talle:

XXL

Formularios – Ejemplo con `input type="radio"` y `"checkbox"`

- Otra forma de trabajar, es crear un arreglo de checkbox

```
<form action="php/compra.php" method="post">
  <fieldset>
    <legend>Tipo de Prenda</legend>
    <label for="cami">Camisa</label>
    |   <input type="checkbox" value="Camisa" name="prenda[]" id="cami">
    <label for="reme">Remera</label>
    |   <input type="checkbox" value="Remera" name="prenda[]" id="reme">
    <label for="saco">Saco</label>
    |   <input type="checkbox" value="Saco" name="prenda[]" id="saco">
    <label for="camp">Campera</label>
    |   <input type="checkbox" value="Campera" name="prenda[]" id="camp">
  </fieldset>
```



Formularios – Ejemplo con `input type="radio"` y `"checkbox"`

- Para recorrer el arreglo de prendas usaremos `foreach`

```
pp20 > htdocs > clase04 > 10-prendas-radio-check2 > php > compra.php
```

```
<?php
    // se debería incluir un encabezado
    var_dump($_POST); // ver qué índices tiene, por lo general no se muestra
    // controlar que los elementos no estén vacíos
    if (!empty($_POST['prenda']) && !empty($_POST['talle'])) {
        echo '<h2>Tipo de prenda:</h2>';
        foreach ($_POST['prenda'] as $prenda) {
            echo '<p>' . $prenda . '</p>';
        }
        echo '<h2>Talle: </h2>';
        echo '<p>' . $_POST['talle'] . '</p>';
    } else {
        echo '<p>Debe seleccionar al menos una prenda y el talle</p>';
    }
?>
```


Formularios – Ejemplo con `input type="radio"` y `"checkbox"`

- Observe cómo se generó el arreglo prenda

← → ↻ ⓘ localhost/clase04/10-prendas-radio-check2/php/compra.php

```
array(2) { ["prenda"]=> array(3) { [0]=> string(6) "Camisa" [1]=> string(6) "Remera" [2]=> string(4) "Saco" } ["talle"]=> string(3) "XXL" }
```

Tipo de prenda:

Camisa

Remera

Saco

Talle:

XXL

Formularios – Ejemplo con textarea

- Crear un formulario de consultas, que tenga un campo texto para el título, uno para correo y un textarea.
- Otra página llamada consulta.php, recibirá los datos, verificará que no estén vacíos y mostrará la consulta realizada.

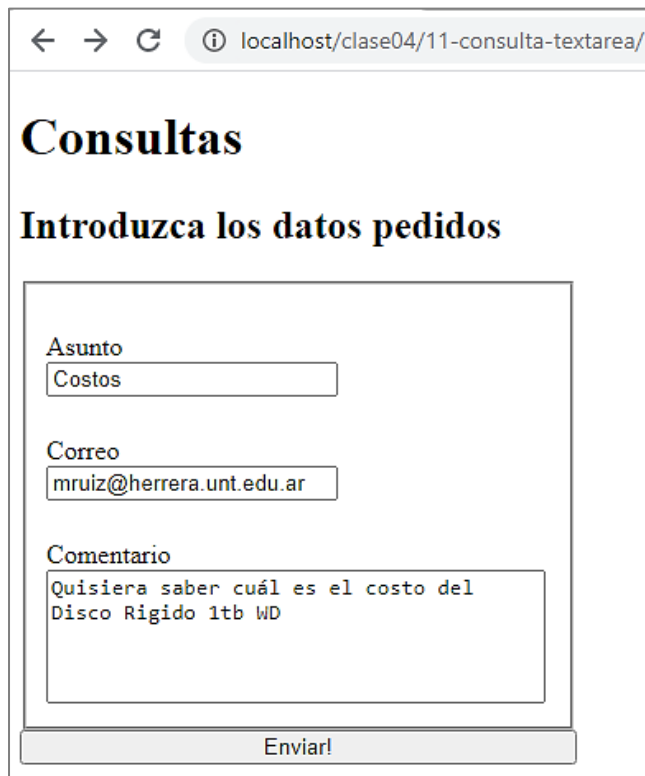
Formularios – Ejemplo con textarea

Index.html

```
<article>
  <h2>Introduzca los datos solicitados</h2>
  <form action="php/consulta.php" method="post">
    <fieldset>
      <label for="asu">Asunto</label>
      <input type="text" name="asunto" id="usu" required>
      <label for="cor">Correo</label>
      <input type="email" name="correo" id="cor" required>
      <label for="comen">Comentario</label>
      <textarea name="comentario" id="comen" cols="40" rows="5" required>
      </textarea>
    </fieldset>
    <input type="submit" value="Enviar!">
  </form>
```

Formularios – Ejemplo con `textarea`

Index.html



A screenshot of a web browser window. The address bar shows 'localhost/clase04/11-consulta-textarea/'. The page has a title 'Consultas' and a subtitle 'Introduzca los datos pedidos'. Below these are three form fields: 'Asunto' with the value 'Costos', 'Correo' with the value 'mruiz@herrera.unt.edu.ar', and 'Comentario' with the text 'Quisiera saber cuál es el costo del Disco Rígido 1tb WD'. At the bottom is a button labeled 'Enviar!'.

← → ↻ ⓘ localhost/clase04/11-consulta-textarea/

Consultas

Introduzca los datos pedidos

Asunto
Costos

Correo
mruiz@herrera.unt.edu.ar

Comentario
Quisiera saber cuál es el costo del
Disco Rígido 1tb WD

Enviar!

Formularios – Ejemplo con textarea

consulta.php

```
<?php
    // se debería incluir un encabezado si se va a mostrar
    // sino, los datos podrían guardarse en una Base de datos
    // controlar que los elementos no estén vacíos
    if (!empty($_POST['asunto']) && !empty($_POST['correo']) && !empty($_POST['comentario'])) {
        echo '<h2>Asunto:</h2>';
        echo '<p>' . $_POST['asunto'] . '</p>';
        echo '<h2>Correo: </h2>';
        echo '<p>' . $_POST['correo'] . '</p>';
        echo '<h2>Comentario: </h2>';
        echo '<p>' . $_POST['comentario'] . '</p>';
    } else {
        echo '<p>Faltaron llenar campos</p>';
    }
?>
```

Formularios – Ejemplo con textarea

consulta.php



A screenshot of a web browser window displaying a form. The address bar shows the URL 'localhost/clase04/11-consulta-textarea/php/consulta.php'. The form contains three sections: 'Asunto:' with the text 'Costos', 'Correo:' with the email 'mruiz@herrera.unt.edu.ar', and 'Comentario:' with the text 'Quisiera saber cuál es el costo del Disco Rígido 1tb WD'.

← → ↻ ⓘ localhost/clase04/11-consulta-textarea/php/consulta.php

Asunto:

Costos

Correo:

mruiz@herrera.unt.edu.ar

Comentario:

Quisiera saber cuál es el costo del Disco Rígido 1tb WD

Archivos con funciones - Ejemplo

- Crear un archivo llamado **index.html**, que tenga un formulario que permita ingresar 3 notas, cuya limitación sea que tengan decimales y estén entre 0 y 10. Dicho formulario enviará las notas al archivo operar.php.
- Crear un archivo llamado **funciones.php**, en este crear las Funciones:
 - **promedio**, que recibirá 3 notas y retornará el promedio de las mismas.
 - **maximo**, que recibirá 3 notas y retornará cual es la mayor nota.
 - **mostrar**, que recibirá el promedio y maximo calculado, y cuya tarea será mostrarlos en la página.
- El archivo **operar.php**, se encargará de chequear la recepción de las notas y luego llamará a las distintas funciones del archivo funciones.php

Inclusión de archivos externos - Esquema Ejemplo



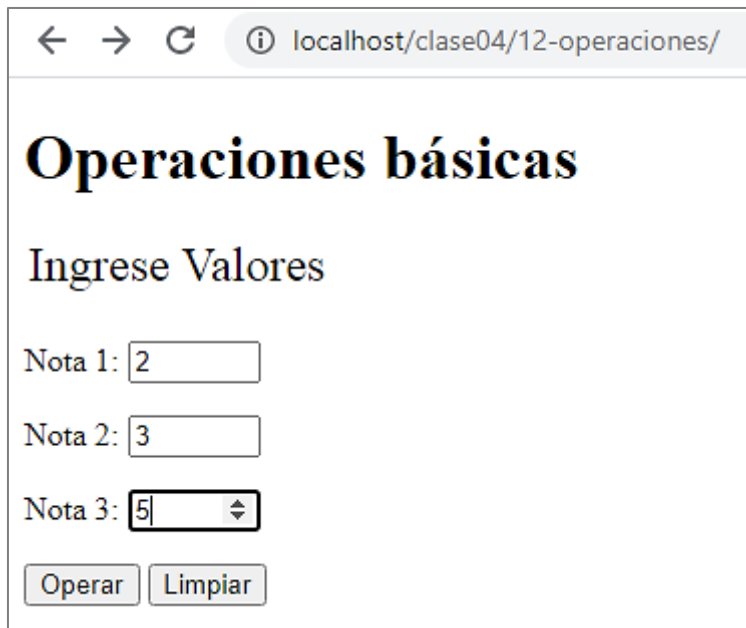
Inclusión de archivos con funciones - Ejemplo

- Formulario en html

```
<article>
  <form action="php/operar.php" method="post">
    <legend>Ingrese Valores</legend>
    <label for="num1">Nota 1:
      <input type="number" name="nota1" id="num1" min="0" max="10">
    </label>
    <label for="num2">Nota 2:
      <input type="number" name="nota2" id="num2" min="0" max="10">
    </label>
    <label for="num3">Nota 3:
      <input type="number" name="nota3" id="num3" min="0" max="10">
    </label>
    <input type="submit" value="Operar">
    <input type="reset" value="Limpiar">
  </form>
</article>
```

Inclusión de archivos con funciones - Ejemplo

- Formulario en html



A screenshot of a web browser window displaying a form titled "Operaciones básicas". The browser's address bar shows the URL "localhost/clase04/12-operaciones/". The form contains the instruction "Ingrese Valores" followed by three input fields labeled "Nota 1:", "Nota 2:", and "Nota 3:". The first two are text inputs with values "2" and "3" respectively. The third is a dropdown menu with "5" selected. At the bottom are two buttons: "Operar" and "Limpiar".

← → ↻ ⓘ localhost/clase04/12-operaciones/

Operaciones básicas

Ingrese Valores

Nota 1:

Nota 2:

Nota 3:

Inclusión de archivos con funciones - Ejemplo

- funciones.php

```
<?php
function promedio ($num1, $num2, $num3)
{
    $prom = ($num1 + $num2 + $num3)/3;
    return $prom;
}

function maximo ($num1, $num2, $num3)
{
    if (($num1 > $num2) && ($num1 > $num3)) {
        $max = $num1;
    } elseif (($num2 > $num1) && ($num2 > $num3)) {
        $max = $num2;
    } elseif (($num3 > $num1) && ($num3 > $num2)) {
        $max = $num3;
    } elseif (($num1 == $num2) || ($num1 == $num3) || ($num2 == $num3) ) {
        $max = -99; // hay igualdad, no se pudo identificar el mayor
    }
    return $max;
}
```

Inclusión de archivos con funciones - Ejemplo

- funciones.php (continuación...)

```
        return $max;
    }

    function mostrar ($num1, $num2)
    {
        echo '<p>Promedio: ' . number_format($num1, 2., ',', '.') . ' y Máximo: ' . $num2 . '</p>';
    }
?>
```

Inclusión de archivos con funciones - Ejemplo

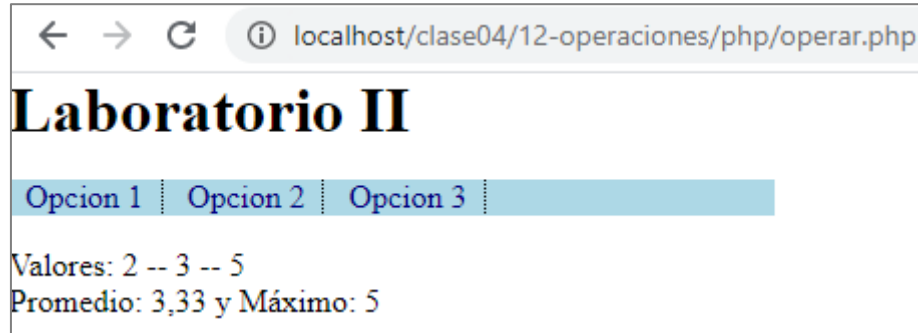
- operar.php

Es necesario incluir el archivo para poder usar las funciones.

```
<?php
require_once '../html/encabezado.html';
require_once 'funciones.php';
if (!empty($_POST['nota1']) && !empty($_POST['nota2']) && !empty($_POST['nota3'])) {
    $prom = promedio($_POST['nota1'], $_POST['nota2'], $_POST['nota3']); // invoca función
    $n1 = $_POST['nota1'];
    $n2 = $_POST['nota2'];
    $n3 = $_POST['nota3'];
    echo '<p>Valores: ' . $n1 . ' -- ' . $n2 . ' -- ' . $n3 . '</p>';
    $max = maximo($n1, $n2, $n3); // invoca función
    mostrar($prom, $max); // se llama a un procedimiento
}
include '../html/pie.html';
?>
```

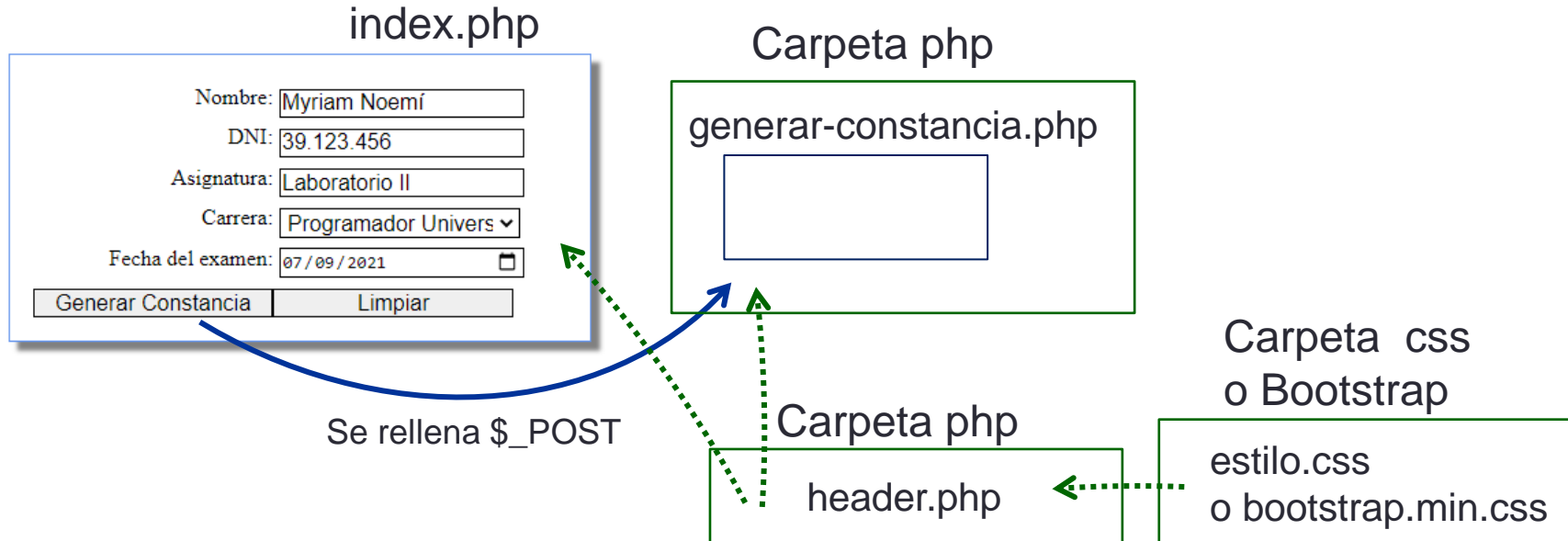
Inclusión de archivos con funciones - Ejemplo

- Lo que muestra operar.php



Inclusión de archivos css en distintas ubicaciones


- Realizar un formulario que envíe los datos a generar-constancia.php y que utilice **un único archivo de header.php** en este y en index



Inclusión de archivos css en distintas ubicaciones

- El archivo header.php, tendrá una variable que le permitirá cambiar la ubicación del CSS, a fin de que funcione en cualquier ubicación del archivo php.

```
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="author" content="Myriam Ruiz">
    <meta name="description" content="Liquidación de Sueldos">
    <link rel="stylesheet" href="<?php echo $ruta;?>/estilo.css">
    <title>Ejemplo Función y Formulario</title>
</head>
<body>
```



Inclusión de archivos css en distintas ubicaciones

- En cada archivo php se podrá indicar la ubicación del css con respecto a si mismo (ej: index.php):

```
<?php
    $ruta = 'css'; ←
    require_once 'php/header.php';
?>
<form action="php/generar-constancia.php" method="post">
    <label for="nom">Nombre:</label>
    <input type="text" name="nombre" id="no">
    <label for="docu">DNI:</label>
    <input type="text" name="dni" id="docu">
    <label for="asig">Asignatura:</label>
    <input type="text" name="asignatura" id="asig">
```

Inclusión de archivos css en distintas ubicaciones

- En cada archivo php se podrá indicar la ubicación del css con respecto a si mismo (ej: generar-constancia.php):

```
<?php
    $ruta = '../css'; ←
    include '../php/header.php';
    if (!empty($_POST['fecha']) && !empty($_POST['nombre'])
        && !empty($_POST['dni']) && !empty($_POST['carrera'])
        && !empty($_POST['asignatura'])) {
        $fecha = $_POST['fecha'];
        $nombre = $_POST['nombre'];
        $dni = $_POST['dni'];
        $asignatura = $_POST['asignatura'];
        $carrera = $_POST['carrera'];
    }
?>

<p class="fecha">San Miguel de Tucumán, <?php echo $fecha;?></p>
```

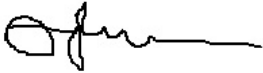
Inclusión de archivos css en distintas ubicaciones

- Luego de controlar que tenemos todos los datos, se arma la constancia:

```
<p class="fecha">San Miguel de Tucumán, <?php echo $fecha;?></p>
<p>
----Por la presente, el Secretario de Bienestar Estudiantil de la Facultad de Ciencias
Exactas y Tecnología de la Universidad Nacional de Tucumán, Ayarde Fabián Fernando,
deja expresa CONSTANCIA que, el/la alumno/a: <strong><?php echo $nombre;?> - DNI: <?php echo
$dni;?></strong>, rindió el <?php echo $fecha;?> un examen Parcial de la asignatura
"<?php echo $asignatura;?>", perteneciente a la carrera de <?php echo $carrera;?> de esta
Facultad.</p>
<p>
----Para ser presentada ante quien corresponda, se expide la presente en San Miguel de
Tucumán, en el día de la fecha.- </p>
<figure>
    
    <figcaption>Fabián Fernando Ayarde <br>Secretario de Bienestar Estudiantil</figcaption>
</figure>
```

Inclusión de archivos css en distintas ubicaciones

- Resultado final

Constancia de Examen
<p style="text-align: right;">San Miguel de Tucumán, 2021-09-07</p> <p>----Por la presente, el Secretario de Bienestar Estudiantil de la Facultad de Ciencias Exactas y Tecnología de la Universidad Nacional de Tucumán, Ayarde Fabián Fernando, deja expresa CONSTANCIA que, el/la alumno/a: Myriam Noemi - DNI: 39.123.456, rindió el 2021-09-07 un examen Parcial de la asignatura "Laboratorio II", perteneciente a la carrera de Programador Universitario de esta Facultad.</p> <p>----Para ser presentada ante quien corresponda, se expide la presente en San Miguel de Tucumán, en el día de la fecha.-</p> <div style="text-align: right;"> Fabián Fernando Ayarde Secretario de Bienestar Estudiantil</div>

(el formato de la fecha se trabajará más adelante)

Comprobación Teórica 04

<https://tinyurl.com/Q4L-2024>

