UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES
Centro de e-Learning
Secretaría de Cultura y Extensión Universitaria

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Curso:

PROGRAMADOR WEB INICIAL

UTN.BA | Centro de e-Learning
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES
Secretaría de Cultura y Extensión Universitaria

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Módulo 3:

PHP Y JAVASCRIPT

Unidad 1:

INTRODUCCIÓN A PHP

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Presentación

En esta unidad, marcaremos la diferencia entre las páginas estáticas y las páginas dinámicas. Seguidamente, nos vamos a introducir al lenguaje de programación PHP. Veremos cómo funciona, cómo configurar el entorno de trabajo y cómo emplearlo en nuestros proyectos. Para ello, conoceremos cuál es su sintaxis y cómo se intercala dentro del código HTML.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Objetivos

Que los participantes logren...

- Conocer las posibilidades de configuración de PHP.
- Introducirse en su funcionamiento.
- Emplear la sintaxis propia del lenguaje e intercalar las instrucciones dentro del código HTML.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Bloques temáticos

1. Conceptos iniciales de PHP
    a. Concepto de páginas dinámicas
2. ¿Qué es PHP?
    a. ¿Dónde puede ser utilizado PHP?
3. ¿Cómo funciona PHP?
4. Sintaxis en PHP
5. Variables en PHP
    a. Reglas para variables de PHP
    b. Tipos de datos
6. Formularios
    a. Variable $_GET
    b. Variable $_POST
7. Función mail

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

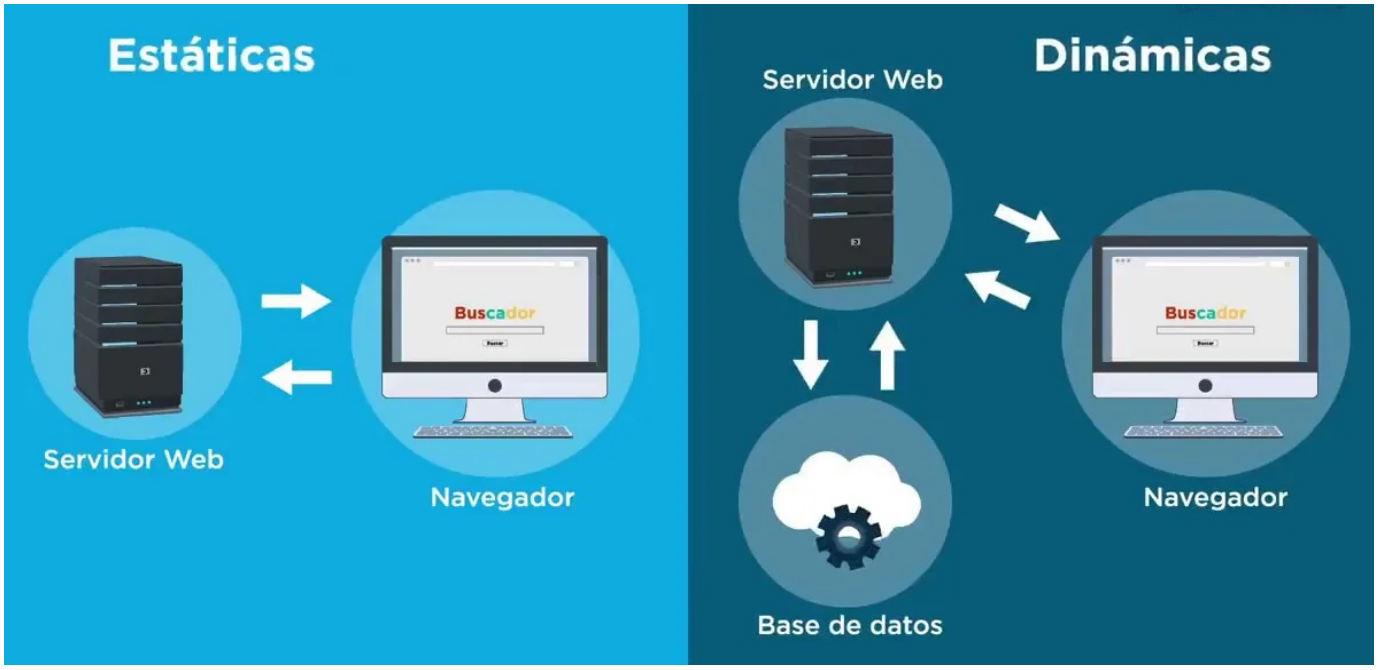
Conceptos iniciales de PHP

Concepto de páginas dinámicas

Hasta hace poco tiempo, la mayoría de las páginas web eran estáticas: páginas donde hay información que no cambia y se muestra de manera permanente. Están desarrolladas principalmente con HTML y CSS. Su aspecto puede ser muy parecido al de una web dinámica, pero a la hora de actualizarlas, aunque sea una mínima alteración, deberá realizarse directamente en el código fuente. Por lo tanto, no pueden ser mantenidas fácilmente por una persona que no tenga conocimientos de programación.

Actualmente, la mayoría de las páginas web son dinámicas porque se construyen utilizando otros lenguajes de programación como PHP, lo que permite agregar aplicaciones para muchos tipos de funcionalidades, como: blogs, foros, tiendas online, entre otras.

La característica principal es que el contenido se puede modificar fácilmente. Es mucho más fácil añadir contenido y modificar cualquier elemento de la web. Utilizaremos lenguajes que poseen scripts que nos ayudarán a automatizar determinadas tareas en nuestras páginas web



¿Qué es PHP?

PHP (Hypertext Preprocessor o traducido como Procesador Previo al Hipertexto) es un lenguaje de código abierto muy popular para el desarrollo web y que puede ser incrustado directamente en HTML. Veámoslo en el siguiente ejemplo:

```
<!DOCTYPE html>
<html>
<head>
	<title>Ejemplo</title>
</head>
<body>
	<?php
		echo "Este es un script de PHP";
	?>
</body>
</html>
```

El código de PHP se encuentra encerrado entre las etiquetas especiales de comienzo **<?php** y final **?>** que marcan el bloque de texto correspondiente a PHP.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Este lenguaje nos permitirá utilizarlo junto a Bases de Datos, brindándonos respuestas dinámicas a los diferentes scripts que utilizaremos en nuestros sitios, y será el intérprete del servidor web, quien se encargará de procesarlos.

¿Dónde puede ser utilizado PHP?

Con PHP, tenemos la libertad de elegir el sistema operativo y el servidor web en el que deseamos trabajar. PHP puede ser utilizado en una amplia variedad de sistemas operativos principales, incluyendo Linux, varias variantes de Unix, Microsoft Windows, macOS, RISC OS, entre otros. PHP es compatible con la mayoría de los servidores web utilizados en la actualidad, como Apache, IIS y otros. Además, podemos optar por emplear programación por procedimientos, programación orientada a objetos (POO), o una combinación de ambas metodologías.

Es importante destacar que PHP no se limita únicamente a la generación de código HTML; también permite la creación de imágenes y archivos PDF, así como el procesamiento de diversos tipos de texto y formatos de ficheros XML.

La integración de PHP con sistemas de gestión de bases de datos nos capacita para desarrollar una amplia gama de aplicaciones, tales como foros, blogs, calendarios, sistemas de autenticación con usuarios y contraseñas para páginas específicas de nuestro sitio, así como formularios con envío de correos electrónicos automáticos, entre otras funcionalidades.

PHP fue creado en 1994 por Rasmus Lerdorf con la intención inicial de construir un contador y un libro de visitas para rastrear las visitas a su currículum en línea. Desde entonces, PHP ha experimentado una rápida evolución, siendo adoptado por una amplia comunidad de desarrolladores que han enriquecido el lenguaje con nuevas funciones y características útiles para la construcción de sitios web.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

En la actualidad, la última versión estable de PHP es la 8.x. Sin embargo, es importante mencionar que la mayoría de los sitios web y servidores utilizan versiones anteriores, como 7.2 o 7.3, debido a la necesidad de mantener la compatibilidad con aplicaciones existentes y asegurar la estabilidad del entorno.

Es crucial destacar que la tecnología Flash, que solía utilizarse para crear películas y animaciones interactivas en la web, ha quedado obsoleta y ya no es compatible con la mayoría de los navegadores modernos. En lugar de Flash, se han adoptado tecnologías más seguras y eficientes, como HTML5, CSS3 y JavaScript, para la reproducción de contenido multimedia en línea.

¿Cómo funciona PHP?

PHP se ejecuta en un servidor web, por lo que debe tener un servidor web local o remoto para ejecutarlo.
En la práctica, PHP realiza tareas complejas del lado del servidor y, en la mayoría de los casos, genera código HTML que un navegador web interpreta y muestra.

Normalmente, las aplicaciones web en PHP realizan tareas complejas como la interoperabilidad con bases de datos, seguridad, sistemas de inicio de sesión y registro de usuarios, gestión de plantillas de maquetación o "themes", envío de correos electrónicos, procesamiento de formularios, gestión de archivos y directorios, cookies y comunicación con otros sistemas. a través de interfaces REST en la "API" y otras tareas.

Sintaxis en PHP

- Las sentencias siempre finalizan con **un punto y coma (;).**
- El código PHP siempre debe estar encerrado por sus delimitadores de inicio y final (**<?php** y **?>**).
- Los comentarios de una sola línea en PHP pueden hacerse empezando la línea con **//** o con **#.**
- Los comentarios de múltiples líneas están encerrados entre: **/*** y ***/.**
- PHP es un lenguaje que distingue entre mayúsculas y minúsculas, "VAR" no es lo mismo que "var".

Variables en PHP

Recordemos que las variables son "contenedores" para almacenar información que podremos utilizar cuantas veces queramos. En PHP declararemos las variables anteponiendo el signo **$** al nombre elegido.

Ejemplo

```
<?php
        $ = "Este es un texto de ejemplo!";
        $x = 5;
        $y = 10.5;
?>
```

Después de la ejecución de las instrucciones anteriores, las variables almacenarán los siguientes valores:

- $texto: Este es un texto de ejemplo! (Debemos recordar que si estamos ingresando un valor de tipo string, debemos colocar comillas)
- $x: el valor 5
- $y: el valor 10.5.

Reglas para variables de PHP

Podremos asignar los nombres a la variables de dos maneras: un nombre corto (como x, y o var1) o un nombre descriptivo (nombre, valor_producto, total_compras que nos ayudará a comprender mejor el código en el futuro, o bien si otro desarrollador lee nuestro código.

- Un nombre de variable debe comenzar con una **letra** o el **carácter de subrayado (_)**, siempre seguido del signo **$.**
- **Nunca** puede comenzar con un número.
- Solo puede contener caracteres alfanuméricos y guiones bajos (Az, 0-9 y _)
- Distinguen entre mayúsculas y minúsculas, por lo tanto $edad y $EDAD son dos variables distintas.

Por ejemplo:

```
<?php
        $var = 'Fernando';
        $Var = 'Eva';
        echo "$var, $Var";      // imprime "Fernando, Eva"

        $4site = 'aun no';      // inválido; comienza con un número
```

```
        $_4site = 'aun no';     // válido; comienza con un carácter de subrayado
        $täyte = 'mansikka';    // válido; 'ä' es ASCII (Extendido) 228
?>
```

Tipos de datos

Las variables pueden almacenar datos de diferentes tipos, y diferentes tipos de datos pueden hacer cosas diferentes. PHP admite los siguientes tipos de datos:

- **String**
  Una cadena es una secuencia de caracteres, como "¡Hola mundo!". Una cadena puede ser cualquier texto entre comillas. Puede utilizar comillas simples o dobles
- **Enteros**
  Un tipo de datos entero es un número no decimal entre -2,147,483,648 y 2,147,483,647.

  Reglas para números enteros:
  - Un número entero debe tener al menos un dígito.
  - Un número entero no debe tener un punto decimal.
  - Un entero puede ser positivo o negativo.
  - Los números enteros se pueden especificar en: notación decimal (base 10), hexadecimal (base 16), octal (base 8) o binaria (base 2).

- **Float** (números de coma o punto flotante, también llamados dobles)
  Un float es un número con un punto decimal o un número en forma exponencial.
- **Booleano**
  Un booleano representa dos estados posibles: VERDADERO o FALSO. Los booleanos se utilizan a menudo en pruebas condicionales.

  $x = true;

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

```
$y = false;
```

- **Array**
  Una matriz almacena múltiples valores en una sola variable.
- **Objeto**
- **NULL**
  Es un tipo de datos especial que solo puede tener un valor: NULL. Una variable de tipo de datos NULL es una variable que no tiene ningún valor asignado. Si se crea una variable sin un valor, se le asigna automáticamente un valor NULL. Las variables también se pueden vaciar estableciendo el valor en NULL.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Formularios

Hoy en día, los formularios son unos de los elementos más básicos de las estructuras HTML. Su comportamiento y funcionalidad se han convertido en un aspecto muy importante del sitio web, ya que nos permiten crear interacciones con los usuarios y obtener información o contenido que nos permiten prestar un servicio determinado.

En primer lugar, debemos establecer los requisitos básicos para poder realizar los formularios. Debemos recordar que para poder procesar archivos PHP, como páginas web, debemos tener un servidor web habilitado o un servicio activado en uno de nuestros equipos.

Para ello basta con instalar Apache y realizar las conexiones necesarias con el intérprete de PHP, y si estamos utilizando una máquina Windows, instalar un programa como XAMPP o WAMP para ejecutar todos los servicios e incluso bases de datos.

Ejemplo :Un formulario HTML sencillo

```
<form action="accion.php" method="post">
 <p>Nombre: <input type="text" name="nombre" /></p>
 <p>Edad: <input type="text" name="edad" /></p>
 <p><input type="submit" /></p>
</form>
```

No hay nada especial en este formulario. Es solo un formulario HTML sin etiquetas especiales. Cuando el usuario complete este formulario y haga clic en Enviar, se llamará a la página accion.php. En este archivo podemos escribir algo como esto:

```
Hola <?php echo htmlspecialchars($_POST['nombre']); ?>.
Su edad es <?php echo (int)$_POST['edad']; ?> años.
```

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Un ejemplo del resultado de este script podría ser:

```
Hola Hernán. Su edad es 34 años.
```

Variable $_GET

Es una matriz asociativa de variables pasadas a la secuencia del script actual con los parámetros de URL (también conocida como cadena de consulta). Tengamos en cuenta que esta tabla se completa no solo para las solicitudes GET, sino también para todas las solicitudes con una cadena de consulta.

Cuando vemos un signo de interrogación después de la URL del documento, corresponde al nombre de la variable y el valor correspondiente. Luego de la segunda, a cada variable en lugar de un signo de interrogación se le coloca el símbolo ampersand (&) adelante.

Variable $_POST

$_POST es otra matriz asociativa que también almacena información que se pasa de lado a lado como una variable $_GET, pero en este caso almacena información que pasa por otro método, los formularios.

La diferencia con $_GET es que las variables enviadas a "post" no aparecerán en la URL de la página a la que se envían los datos, por lo que es más conveniente usar el método "post" en lugar de "get" en formularios.

Función mail

La función mail() permite enviar correos electrónicos directamente desde el script de PHP. Para utilizarla, PHP requiere que un sistema de correo electrónico esté instalado y funcionando. El programa a usar está determinado por los ajustes de configuración en el archivo php.ini. Esta función viene incorporada en PHP., no se requiere instalar nada para utilizarla. Su sintaxis es la siguiente:

```
mail(to,subject,message,headers,parameters);
```

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Parámetro		Descripción
to	Requerido	Especifica el destinatario o destinatarios del correo electrónico
subject	Requerido	Especifica el asunto del correo electrónico. Nota: este parámetro no puede contener ningún carácter de nueva línea
message	Requerido	Define el mensaje a enviar. Cada línea debe estar separada por un LF (\n). Las líneas no deben exceder los 70 caracteres.
headers	Opcional	Especifica encabezados adicionales, como De, CC y CCO. Los encabezados adicionales deben separarse con un CRLF (\r\n).
parameters	Opcional	Especifica un parámetro adicional para el programa sendmail (el definido en la opción de configuración sendmail_path). (es decir, esto se puede usar para configurar la dirección del remitente del mail cuando se usa sendmail con la opción -f sendmail)

Realizaremos un ejemplo enviando un correo electrónico con encabezados adicionales:

```
<?php
$to = "destinatario@gmail.com";
$subject = "Este es el asunto del mail";
$txt = "Acá estamos escribiendo el texto del email";
$headers = "From: remitente@gfrba.utn.edu.ar" . "\r\n" .
"CC: mailrespaldo@gmail.com";

mail($to,$subject,$txt,$headers);
```

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

```
?>
```

En resumen

En esta unidad, aprendimos cómo otorgarle dinamismo a nuestras web haciendo uso de PHP. Nos introdujimos al lenguaje y conocimos las principales tareas que podemos desarrollar con él.

Hemos trabajado con variables, tipos de datos, variable $_GET y $_POST e integramos los conceptos otorgándole alguna funcionalidad a un formulario.

Bibliografía utilizada y sugerida

- PHP - News Archive - 2021. Recuperado de: https://www.php.net/archive/2021.php
- PHP. Tratar con formularios. Recuperado de: https://www.php.net/manual/es/tutorial.forms.php
- W3Schools. Manejo de formularios PHP. Recuperado de: https://www.w3schools.com/php/php_forms.asp