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 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

Módulo 3:

PHP Y JAVASCRIPT

Unidad 3:

PHP y MySQL

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 aprenderemos a trabajar con arrays. Un array (en inglés), vector, formación o matriz (en español) puede entenderse como una zona de almacenamiento de datos en donde se asocia valores con claves.

También, abordaremos el concepto de función y aprenderemos a conectarnos con la base de datos.

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 qué es y cómo funciona un array o vector.
- Comprender el concepto de función.
- Emplear funciones en PHP para MySQL.

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. Arrays en PHP
    a. ¿Qué es un array?
    b. Crear una matriz en PHP
        i. Matrices indexadas
        ii. Matrices asociativas
        iii. Matrices multidimensionales
2. Funciones
3. Extensión MySQLi
    a. ¿Qué es la extensión mysqli de PHP?
    b. Funciones de la extensión MySQLi
        i. Función mysqli_connect()
        ii. Función mysqli_query()
        iii. Función mysqli_fetch_array()
        iv. Función mysqli_free_result()
        v. Función mysqli_error()
        vi. Función mysqli_close()

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

Arrays en PHP

Las matrices, o también llamados arrays o vectores, almacenan múltiples valores en una variable. Por ejemplo, si desea almacenar los platos de un menú, en lugar de definir cada plato en variables, definiremos una matriz.

```
<?php
	$platos = array("carne asada", "fideos", "pizza");
	echo "En el menú podremos ver " . $platos[0] . ", " . $platos[1] . " y "
	. $platos[2] . ".";
?>
```

Nos devolverá lo siguiente:

```
En el menú podremos ver carne asada, fideos y pizza.
```

¿Qué es un array?

Los array son un tipo especial de variable que puede almacenar múltiples valores a la vez. Si tiene una lista de elementos (en el ejemplo, una lista de platos de un menú), el almacenamiento de los platos en diferentes variables podría verse así:

```
$plato1 = "carne asada";
$plato2 = "fideos";
$plato3 = "pizza";
```

Si quisiéramos recorrer a través de los platos y encontrar uno específico, pero en vez de tener 3 platos, son 30 o 100, podremos crear un array o matriz.

Una matriz puede contener múltiples valores bajo **un solo nombre** al que se puede acceder por referencia a algún índice.

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

Crear una matriz en PHP

En PHP, la función array() se usa para crear matrices:

```
array();
```

Hay tres tipos de arrays en PHP:

- **Matriz indexada** - una matriz con índice numérico. Los valores se almacenan y se accede a ellos de forma lineal.
- **Matriz asociativa** - una matriz con claves nombradas. Estas almacenan los valores de los elementos en relación con los valores clave, no en el orden estricto del índice lineal.
- **Matriz multidimensional** - una matriz contiene una o más matrices y se puede acceder a los valores mediante múltiples índices.

Matrices indexadas

Estas matrices pueden contener números, cadenas y cualquier otro objeto, pero sus índices estarán representados por números. De forma predeterminada, los índices de matriz comienzan en 0(cero).

Una matriz indexada se puede crear de dos maneras:

- Los índices se pueden asignar automáticamente (los índices siempre comienzan en 0) de la siguiente manera:

```
$platos = array("carne asada", "fideos", "pizza");
```

- Por otro lado, el índice se puede especificar manualmente:

```
$plato[1] = "carne asada";
$plato[2] = "fideos";
$plato[3] = "pizza";
```

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

Matrices asociativas

Los arrays asociativos son vectores que usan claves con el nombre específico que le asignemos. Son muy similares en funcionalidad a las matrices con índice numérico, pero difieren en su indexación. Las matrices asociativas tienen sus índices en forma de cadena, por lo que puede crear asociaciones sólidas entre claves y valores.

Una tabla indexada numéricamente no es la mejor solución para almacenar los salarios de los empleados en la tabla. En cambio, podríamos usar los nombres de cada empleado como clave en nuestra matriz y el valor podría ser su salario.

Hay dos formas de crear una matriz asociativa:

```
$edad = array("Maria" => "28", "Jorge" => "32", "Florencia" => "22");
```

o:

```
$edad[Maria] = "28";
$edad[Jorge] = "32";
$edad[Florencia] = "22";
```

Las claves con nombre se pueden usar en scripts, veamos un ejemplo:

```
<!DOCTYPE html>
<html>
	<body>
		<?php
		$edad = array("Maria"=>"28", "Jorge"=>"32", "Florencia"=>"22");
		echo "Maria tiene " . $edad[Maria] . " años.";
		?>
	</body>
</html>
```

Nos devolverá lo siguiente:

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

```
Maria tiene 28 años.
```

Matrices multidimensionales

Una matriz multidimensional es una matriz que contiene una o más matrices.

Hemos descrito una matriz como una lista de pares clave/valor. Sin embargo, a veces desea almacenar valores con varias claves. Para ello disponemos de arrays multidimensionales. Cualquier elemento de la matriz principal puede ser una matriz. Cada miembro de un subconjunto puede ser un conjunto, y así sucesivamente.

PHP admite matrices multidimensionales con una profundidad de dos, tres, cuatro, cinco o más niveles. Sin embargo, para la mayoría de las personas, administrar arreglos de más de tres pisos es difícil.

En el siguiente ejemplo, crearemos una matriz bidimensional para almacenar las calificaciones de tres estudiantes en tres materias:

```
<html>
   <body>
      <?php
         $notas = array(
            "nicolas" => array (
               "lengua" => 8,
               "matematica" => 6,
               "fisica" => 5
            ),

            "ezequiel" => array (
               "lengua" => 6,
               "matematica" => 8,
               "fisica" => 9
            ),

            "victor" => array (
               "lengua" => 9,
               "matematica" => 8,
```

```
            "fisica" => 7
        )
    );

    /* Acceder a la matriz y mostrar las notas*/
    echo "Nota de Nicolas en lengua: " ;
    echo $notas['nicolas']['lengua'] . "<br />";

    echo "Nota de Ezequiel en matematica: ";
    echo $notas['ezequiel']['matematica'] . "<br />";

    echo "Nota de Victor en fisica: " ;
    echo $notas['victor']['fisica'] . "<br />";
  ?>

  </body>
</html>
```

Nos devolverá lo siguiente:

```
Nota de Nicolas en lengua: 8
Nota de Ezequiel en matematica: 8
Nota de Victor en fisica: 7
```

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

Funciones

Una función es un conjunto de declaraciones o comandos que nos permiten pasarles variables, también llamados parámetros y obtener un resultado.

Se definen con la palabra **function** seguido por el **nombre de la función** y **paréntesis ()**, que pueden o no contener parámetros. Finalmente la declaración de la función encerrada entre **llaves {}**.

```
<?php
      function mifuncion(){
            instrucciones;
      }
?>
```

Las funciones a menudo se usan para realizar procedimientos que se usan muchas veces, por lo que es muy conveniente realizar una función y luego simplemente llamar a la función correspondiente.

Tanto las variables como las funciones, deben nombrarse sin signos, espacios o caracteres especiales para evitar el riesgo de errores.

Hay dos cosas importantes que deben saber acerca de las funciones:

- Para pasar datos a una función, la función debe aceptar los datos entre paréntesis.
- Debemos usar la declaración **“return”** para que la función devuelva un resultado.

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

Extensión MySQLi

¿Qué es la extensión mysqli de PHP?

La extensión MySQLi, a veces denominada extensión MySQL mejorada, se desarrolló para aprovechar las nuevas características de los sistemas MySQL que ejecutan la versión 4.1.3 en adelante. La extensión mysqli se incluye por defecto a partir de la versión 5 de PHP.

Las principales mejoras de la extensión MySQL son:

- Interfaz dual: procedimental y orientada a objetos
- Soporte para declaraciones preparadas
- Soporte para múltiples declaraciones
- Soporte de transacciones
- Opciones de depuración mejoradas
- Soporte de servidor incorporado

Funciones de la extensión MySQLi

Función mysqli_connect()

Esta función abre una nueva conexión con el servidor MySQL.

```
mysqli_connect(host,username,password,dbname,port,socket);
```

Parámetro	Descripción
host	Opcional. Especifica un nombre de host o una dirección IP
username	Opcional. Especifica el nombre de usuario de MySQL

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

password	Opcional. Especifica la contraseña de MySQL
dbname	Opcional. Especifica la base de datos por defecto que se utilizará
port	Opcional. Especifica el número de puerto para intentar conectarse al servidor MySQL
socket	Opcional. Especifica la toma o tubería con nombre que se utilizará

Ejemplo: Abrir una nueva conexión con el servidor MySQL:

```
<?php
$con = mysqli_connect("localhost","usuario","contraseña","base_datos");
// Verificar conexión
if (mysqli_connect_errno())
  {
  echo "Failed to connect to MySQL: " . mysqli_connect_error();
  }
?>
```

Función mysqli_query()

Esta función realiza una consulta en la base de datos.

```
mysqli_query(connection,query,resultmode);
```

Parámetro	Descripción
connection	Necesario. Especifica la conexión de MySQL a usar
query	Necesario. Especifica la cadena de consulta
resultmode	Opcional. Una constante: • Mysqli_use_result (Use esto si tenemos que recuperar gran cantidad de datos) • Mysqli_store_result (Esta es la opción por defecto)

Ejemplo: Realizar consultas en la base de datos:

```
<?php
$con=mysqli_connect("localhost","usuario","contraseña","base_datos");
// Verificar conexión
if (mysqli_connect_errno())
  {
  echo "Failed to connect to MySQL: " . mysqli_connect_error();
  }
// Realizar consultas
mysqli_query($con,"SELECT * FROM Personas");
mysqli_query($con,"INSERT INTO Personas (Nombre,Apellido,Edad)
VALUES ('Carlos','Perez'25)");
mysqli_close($con);
?>
```

Función mysqli_fetch_array()

Esta función lee una fila de resultados como un array indexado, un array asociativo:, o ambos.

```
mysqli_fetch_array(result,resulttype);
```

Parámetro	Descripción
result	Necesario. Especifica un identificador de conjunto de resultados devuelto por mysqli_query() , mysqli_store_result() o mysqli_use_result()
resulttype	Opcional. Especifica qué tipo de matriz que debe ser producido. Puede ser uno de los siguientes valores: • MYSQLI_ASSOC • MYSQLI_NUM • MYSQLI_BOTH

Ejemplo: Extrae la fila de resultado como un array indexado y como un array asociativo:

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

```php
<?php
$con=mysqli_connect("localhost","usuario","contraseña","base_datos");
// Verificar conexión
if (mysqli_connect_errno())
  {
  echo "Failed to connect to MySQL: " . mysqli_connect_error();
  }
$sql="SELECT Apellido,Edad FROM Personas ORDER BY Apellido";
$result=mysqli_query($con,$sql);
// Array indexado
$row=mysqli_fetch_array($result,MYSQLI_NUM);
printf ("%s (%s)\n",$row[0],$row[1]);
// Array asociativo
$row=mysqli_fetch_array($result,MYSQLI_ASSOC);
printf ("%s (%s)\n",$row["Apellido"],$row["Edad"]);
// Liberar resultados
mysqli_free_result($result);
mysqli_close($con);
?>
```

Función mysqli_free_result()

Libera la memoria asociada a un resultado, Siempre se debe liberar el resultado con mysqli_free_result(), cuando el objeto del resultado ya no es necesario.

```
mysqli_free_result(result);
```

Parámetro	Descripción
result	Necesario. Especifica un identificador de conjunto de resultados devuelto por mysqli_query() , mysqli_store_result() o mysqli_use_result()

Ejemplo: Obtener las filas de una conjunto de resultados, a continuación, liberar la memoria asociada con dicho resultado:

```php
<?php
$con=mysqli_connect("localhost","usuario","contraseña","base_datos");
```

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

```
// Verificar conexión
if (mysqli_connect_errno())
  {
  echo "Failed to connect to MySQL: " . mysqli_connect_error();
  }
$sql="SELECT Apellido,Edad FROM Personas ORDER BY Apellido";
if ($result=mysqli_query($con,$sql))
  {
  // Obtener filas una por una
  while ($row=mysqli_fetch_row($result))
    {
    printf ("%s (%s)\n",$row[0],$row[1]);
    }
  // Liberar resultados
  mysqli_free_result($result);
}
mysqli_close($con);
?>
```

Función mysqli_error()

Devuelve el último mensaje de error para la llamada más reciente a una función de MySQLi que puede haberse ejecutado correctamente o haber fallado.

```
mysqli_error(connection);
```

Parámetro	Descripción
connection	Necesario. Especifica la conexión de MySQL a usar

Ejemplo: Devolver la última descripción de error para la última llamada a la función

```
<?php
$con=mysqli_connect("localhost","usuario","contraseña","base_datos");
// Verificar conexión
```

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

```
if (mysqli_connect_errno())
  {
  echo "Failed to connect to MySQL: " . mysqli_connect_error();
  }
// Realizar una consulta, comprobar si hay error
if (!mysqli_query($con,"INSERT INTO Personas (Nombre) VALUES ('Carlos')"))
  {
  echo("Error description: " . mysqli_error($con));
  }
mysqli_close($con);
?>
```

Función mysqli_close()

Esta función cierra la conexión a la base de datos previamente abierta.

```
mysqli_close(connection);
```

Parámetro	Descripción
connection	Necesario. Especifica la conexión de MySQL para cerrar

Ejemplo: Cerrar una conexión de base de datos previamente abierta:

```
<?php
$con=mysqli_connect("localhost","usuario","contraseña","base_datos");
// Código PHP a ejecutar
mysqli_close($con);
?>
```

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 qué es un array y continuamos trabajando con funciones de PHP.

Conocimos la extensión mysqli para conectarnos con la base de datos y recorrimos las principales funciones que nos ofrece.

Bibliografía utilizada y sugerida

- PHP - News Archive - 2021. Recuperado de: https://www.php.net/archive/2021.php
- PHP: Arrays - Manual. Recuperado de: https://www.php.net/manual/es/language.types.array.php
- W3Schools. PHP Arrays. Recuperado de: https://www.w3schools.com/php/php_arrays.asp
- The MySQLi Extension Function Summary - Manual. Recuperado de: http://php.net/manual/en/mysqli.summary.php