



UNIVERSIDAD  
DE GRANADA



# Diseño y Desarrollo de Sistemas de Información

Grado en Ingeniería Informática

## PRÁCTICAS: PHP y MySQL

©M.J. Rodríguez Sánchez

*Este documento está protegido por la Ley de Propiedad Intelectual ([Real Decreto Ley 1/1996 de 12 de abril](#)).*

*Queda expresamente prohibido su uso o distribución sin autorización del autor.*

Departamento de Ciencias de la  
Computación e Inteligencia Artificial  
<http://decsai.ugr.es>



**PHP:** Lenguaje de programación del lado del servidor. Se pueden comunicar con un SGBD (en nuestro caso MySQL)

**Cliente:** Navegador web, mostrará el resultado de la manipulación de los datos que hacemos en el Backend.

- SERVIDOR LOCAL**

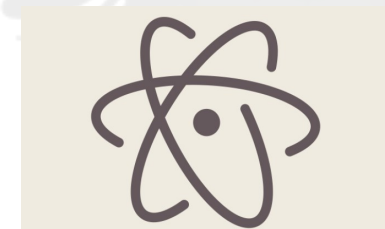


**MAMP**  
manage your website locally

- SGBD**



- IDE**



## Conexión con MySQL desde PHP (3 API)

- **Mysql**
- Desfasada a partir de PHP 5.5
- **Mysqli**
- Mysql optimizado con estructura orientada a objetos. Elimina la necesidad de realizar operaciones adicionales como en mysql.
- A partir de PHP 5.5 y MySQL 4.1.3
- Transacciones a través de la API.
- **PDO**
- Objeto de datos de PHP
- Independientemente de la base de datos que se utilice se emplean las mismas funciones
- Para PHP5+



## Conexión con MySQLi desde PHP

```

1 <?php
2
3 define("DB_HOST","localhost" );
4 define("DB_USER", "root");
5 define("DB_PASS", "");
6 define("DB_DATABASE", "panel" );
7 $db = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_DATABASE);
8
9 if ($db -> connect_errno){ //Si hay algún error en la conexión a la base de datos
10     die("Error de conexión: " . $objetoMysqli->mysqli_connect_errno() . ", " . $objetoMysqli->mysqli_connect_error());
11 }
12 else{
13     echo "La conexión tuvo éxito";
14 }
15
16 $db->mysqli_close(); //Al terminar el script se cierran las conexiones, no hace falta.
17
18 ?>
19

```

**\*Nota: Aconsejo programar la base de datos como una Clase**

## Insertar en BD con MySQLi desde PHP

```

1  <?php
2
3  //Incluyo el archivo donde tengo la base de datos.
4  require_once ('Database.php');
5
6  //Cada tabla una clase
7  class Dominio{
8
9      public function crearDominio($nombre, $id_hosting){
10         $last_id = 0;
11
12         //Conexion BD
13         $db = new Database();
14         $db = $db->connect();
15
16         $sql = "INSERT INTO Dominio (nombre, id_hosting) VALUES ('$nombre', $id_hosting)";
17
18         //Ejecuto la sentencia a la vez que hago el condicional
19         if ($db->query($sql) === TRUE){
20             //Me sirve para ver el último ID autoinc. insertado.
21             $last_id = $db->insert_id;
22             return $last_id;
23         }
24         else{
25             return 0;
26         }
27     }
28 }
29
30 ?>
31

```

## Consulta con MySQLi desde PHP

```

1  <?php
2
3  require_once ("../class/Database.php"); //Con esto incluimos un código externo
4
5  //Conecto a la base de datos
6  $db = new Database();
7  $db = $db->connect();
8
9  //Hago la consulta
10 $sql = "SELECT * FROM Dominio WHERE id=3";
11 $resultado = $db->query($sql); //Ejecuto la consulta
12
13 if (!$resultado){
14     echo "La consulta falló";
15 }
16
17 if($resultado->num_rows == 0){
18     echo "No se han obtenido resultados";
19 }
20 else{
21     //La consulta la manejo como un vector
22     while ($fila = mysqli_fetch_assoc($ingresos_proyecto)) {
23         echo $fila["nombre"]; //Muestro de cada fila lo que me interese de esa consulta.
24     }
25 }
26
27 ?>
28

```

## Borrar con MySQLi desde PHP

```

9
10
11
12
13
14
15
16
17
18
19
20
21
22
23

public function eliminarDominio($id){
    //Conexion BD
    $db = new Database();
    $db = $db->connect();

    $sql = "DELETE from Dominio WHERE id=$id";
    if ($db->query($sql) === TRUE) {
        echo "Se ha borrado correctamente";
    } else {
        echo "Error al borrar: " . $db->error;
    }
}

```



## Sesiones PHP

- Con las variables de sesión en PHP podemos guardar variables comunes en toda nuestra web (como variables globales del sistema).
- Con esto **podemos guardar**, por ejemplo, **usuarios** y datos del mismo.
- Cuando queremos trabajar con sesiones tenemos que poner `session_start()`; **en la primera línea** de la página en cuestión.
- Se pueden guardar muchos valores dentro de la variable, por ejemplo:

```
$_SESSION["nombre"] = "Juan Fuláñez";
$_SESSION["id_usuario"] = 33;
```

- Para comprobar si un usuario está *logueado* podemos comprobar si está la variable establecida con la **función isset**, por ejemplo: `if(isset($_SESSION['nombre'])) { ... }`
- Una vez el usuario haya hecho logout, tendremos que destruir la sesión. Lo haremos con la función `session_destroy()` previamente podremos destruir todas las variables que hayamos guardado inicializando el vector (`$_SESSION = array();`)

## Funciones time() y date()

- Para el manejo de fechas usaremos la función **date()** de PHP.

```
date("d/m/Y") imprime 03/12/2018
```

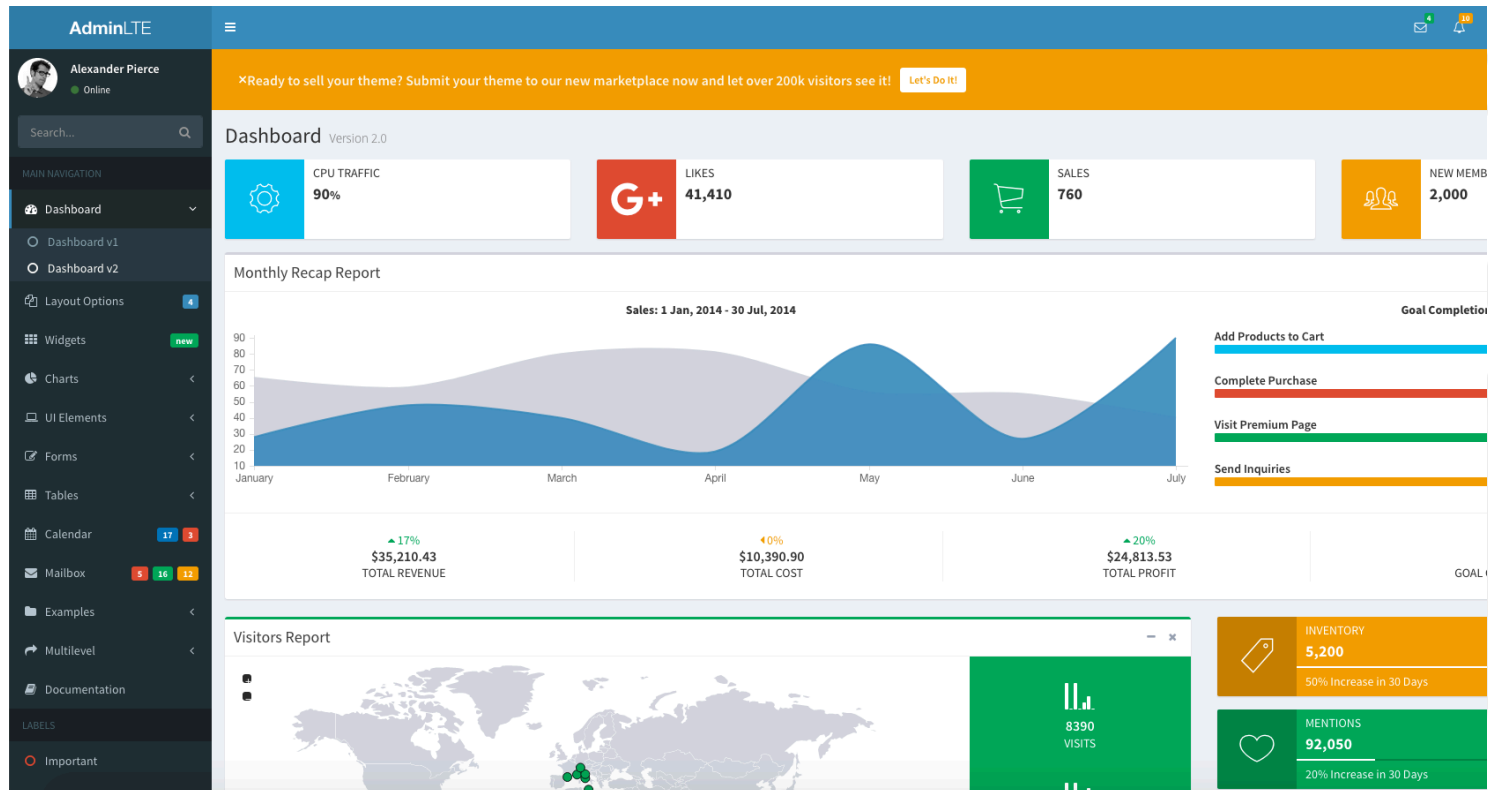
```
date("d/m/Y", strtotime("2018-12-03")) imprime 03/12/2018
```

```
date("d/m/Y", strtotime($timestamp)) imprime la fecha del timestamp
```

- Para obtener el timestamp de la fecha y hora actual utilizaremos la función **time()** de PHP.
- Date coge por defecto como segundo parámetro lo que devuelve time() si no le pasamos nada.
- Más ejemplos:  
<http://php.net/manual/es/function.date.php>  
<http://php.net/manual/es/function.time.php>

## Cliente HTML

<https://adminlte.io/>



## Cliente HTML. Validar formularios con JS.

- Mejora la experiencia del usuario
- Nos permite comprobar el formato de los campos del formulario en la parte del cliente.

```
<form action="" method="" id="" name="" onsubmit="return validacion()">
...
</form>
```

- Entre etiqueta <script> al final del documento antes de cerrar <body>
- Se utilizan expresiones regulares para comprobar por ejemplo si lo introducido es un número, o un email...
- Algún ejemplo: <http://lineadecodigo.com/javascript/validar-que-el-texto-introducido-es-un-numero/> <https://librosweb.es/libro/javascript/capitulo-7/validacion.html>
- Algunas funciones interesantes de Javascript:

```
document.getElementById() /*Selecciona el valor de un elemento*/
alert("Introduzca un valor numérico"); /*Saca una ventana de alerta*/
confirm(mensaje); /*Igual que alert pero obliga a elegir entre 2 opcs.*/
```

## Cliente HTML. Procesar formularios.



- **method="get"**
- Menos seguro
- Datos visibles en URL
- Datos se pueden almacenar en caché

- **method="post"**
- Más seguro
- Recomendado para formularios
- Datos en el cuerpo del documento

## Cliente HTML. Procesar formularios con PHP.

```

14
15 <html>
16 <head><title>Formulario</title></head>
17 <body>
18   <form action="separar.php" method="POST">
19     Nombre:
20     <input type="text" name="nombre"/><br/>
21     Apellidos:
22     <input type="text" name="apellidos" /><br/>
23     <input type="submit" value="Dividir">
24   </form>
25 </body>
26 </html>

```

## Cliente HTML. Procesar formularios con PHP.

```

1  <?php
2  //Cojo los datos del formulario
3  $nombre = $_POST['nombre'];
4  $apellidos = $_POST['apellidos'];
5
6  //Utilizo el . para concatenar
7  echo "El nombre es: ".$nombre." ".$apellidos;
8
9  //Redirecciono
10 header('location: prueba.php');
11
12 ?>
13

```

## Cliente HTML. Procesar formularios con PHP.

```

1  <?php
2
3  if(isset($_POST['enviar'])) {
4      $nombre = $_POST['nombre'];
5      $apellidos = $_POST['apellidos'];
6
7      //Utilizo el . para concatenar
8      echo "El nombre es: ".$nombre." ".$apellidos;
9  }
10
11  ?>
12
13
14  <html>
15  <head><title>Formulario</title></head>
16  <body>
17      <form action="" method="POST">
18          Nombre:
19          <input type="text" name="nombre" /><br/>
20          Apellidos:
21          <input type="text" name="apellidos" /><br/>
22          <input type="submit" name="enviar" value="Dividir">
23      </form>
24  </body>
25  </html>

```



## Cliente HTML. Procesar formularios con PHP.

```

2
3     require_once ('../class/Usuario.php');
4
5     //Paso el parámetro por una url
6     //Por ejemplo editar-usuario.php?id=3
7     $id = $_GET["id"];
8
9     $usuarios = new Usuario();
10    $usuario = $usuarios->consultarUsuario($id);
11
12    if(isset($_POST['enviar'])){
13        $nombre = $_POST['nombre'];
14        $apellidos = $_POST['apellidos'];
15
16        //Utilizo el . para concatenar
17        echo "El nombre es: ".$nombre." ".$apellidos;
18    }
19
20    ?>
21
22    <html>
23    <head><title>Formulario</title></head>
24    <body>
25        <?php
26        while ($fila = mysqli_fetch_assoc($usuario)) {
27            ?>
28
29            <form action="" method="POST">
30                Nombre:
31                <input type="text" name="nombre" value="<?php echo $fila["nombre"] ?>" /><br/>
32                Apellidos:
33                <input type="text" name="apellidos" value="<?php echo $fila["apellidos"] ?>" /><br/>
34                <input type="submit" name="enviar" value="Dividir">
35            </form>
36
37            <?php
38            }
39            ?>

```