

**Tecnólogo en Análisis y Desarrollo de Sistemas de Información**  
**ID 2205627**

**Competencia:** Implantar la solución que cumpla con los requisitos para su operación.

**Resultado de Aprendizaje:** Elaborar el informe administrativo, siguiendo los protocolos de la organización, basado en los planes de instalación, respaldo y migración del sistema de información, facilitando la operatividad y mantenimiento de la solución informática.

**Actividad de Aprendizaje:** 3.1 Programar rutinas de código que gestionen imagines en un BD MYSQL

*La Formación Profesional Integral es el proceso mediante el cual la persona adquiere y desarrolla de manera permanente **conocimientos, destrezas y aptitudes e identifica, genera y asume valores y actitudes** para su realización humana y su participación en el trabajo productivo y en la toma de decisiones sociales. **Acuerdo 12 de 1985 – Unidad Técnico-pedagógica.***

Lo invito a que, de manera dinámica, atenta y propositiva realice el presente instructivo con el ánimo de generar demostrar sus pre saberes en PHP como lenguaje de programación Web.

### Descripción de la Actividad

Estimado aprendiz, esta actividad pertenece a “Apropiación” o también “Mientras aprendo”, tiene como intencionalidad que usted aprenda y practique la mayor cantidad de conocimientos de principios y procesos de la temática trabajada.

#### Programar rutinas de código que gestionen imagines en un BD MYSQL

En este instructivo, realizaremos un ejemplo para subir múltiples imágenes a la base de datos MySQL usando PDO-PHP y las imágenes subidas las mostraremos en la página web.

#### 1. Debemos crear nuestra base de datos....

Es necesario crear una tabla en la base de datos para guardar los nombres de las imágenes. Por lo tanto, las siguientes son las consultas SQL para crear una tabla donde el nombre de la base de datos será “**hv**” y el nombre de la tabla es “**prueba**”.

```

1 CREATE TABLE `prueba` (
2   `id` int(11) NOT NULL,
3   `nombre` varchar(120) COLLATE utf8mb4_unicode_ci NOT NULL,
4   `imagen` varchar(120) COLLATE utf8mb4_unicode_ci NOT NULL
5 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
6
7
8 ALTER TABLE `prueba`
9 ADD PRIMARY KEY (`id`);
10
11 ALTER TABLE `prueba`
12 MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;
```



## 2. Configuración de la cadena de conexión...

Para que el script funcione se debe de conectar a MySQL usando el lenguaje de programación PHP, en este ejercicio usaremos PDO porque es más seguro y robusto.

Crearemos un archivo PHP llamado “cadena.php” y contendrá las variables y tipo de conexión hacia la base de datos.

```
1 <?php
2 try{
3     $base=new PDO('mysql:host=localhost;dbname=hv','root','');
4     //echo "Comunicacion entablada entre PHP y MYSQL...";
5 }catch(Exception $err){
6     die('Error de conexion... '.$err->getMessage());
7 }
8 ?>
```

## 3. Creamos el formulario de registro...

A continuación, crearemos un archivo llamado “registro.php”, que en su estructura HTML mostrara el formulario encargado de cargar las imágenes hacia el servidor y también hacia la base de datos con PHP.

```
53 <form name="frm1" method="POST" action="" enctype="multipart/form-data">
54 <table>
55 <tr>
56 <td colspan="2">...:DATOS DE ASPIRANTE:...</td>
57 </tr>
58 <td>Foto:</td>
59 <td>
60 <input type="file" name="files[]">
61 </td>
62 </tr>
63 <tr>
64 <td colspan="2">
65 <input type="submit" name="btn1" value="ENVIAR">
66 </td>
67 </tr>
68 </table>
69 </form>
```

Es muy importante que la etiqueta `<form>` le agreguemos el parámetro `enctype="multipart/form-data"`, este es el que permite que se suba la imagen, si no se crea este parámetro, los códigos no servirán de nada...

## 4. Código PHP para registrar datos (imágenes)....

Este archivo además del formulario posee una instrucción PHP que se ejecuta al presionar el botón “`btn1`” del formulario. Este botón tiene como tipo “`submit`” en el cual a través de una condicional IF procesara las imágenes, de acuerdo a la siguiente lista:

- A. Contará la cantidad de imágenes
- B. Preparada de la consulta SQL
- C. Generamos el bucle de todos los archivos



INSTRUCTIVO PARA LA EJECUCIÓN DE LA FPI

- D. Extraemos en variable el nombre de archivo
- E. Designamos la carpeta de subida
- F. Obtenemos la extensión del archivo
- G. Validamos la extensión de la imagen
- H. Subimos la imagen al servidor
- I. Ejecutamos la consulta SQL

```
4 <title>Ejercicio imagenes!</title>
5 </head>
6 <?php
7 include 'cadena.php';
8 if(isset($_POST['btn1'])) {
9
10     // Contamos la cantidad de imagenes
11     $countfiles = count($_FILES['files']['name']);
12
13     // Declaración preparada de la consulta SQL
14     $sql1 = "INSERT INTO prueba (nombre,image) VALUES(?,?)";
15
16     $resultado = $base->prepare($sql1);
17
18     // Generamos el bucle de todos los archivos
19     for($i = 0; $i < $countfiles; $i++) {
20
21         // Extraemos en variable el nombre de archivo
22         $filename = $_FILES['files']['name'][$i];
23
24         // Designamos la carpeta de subida
25         $carpeta = 'subidas/'.$filename;
26
27         // Obtenemos la extension del archivo
28         $file_extension = pathinfo($carpeta, PATHINFO_EXTENSION);
29
30         $file_extension = strtolower($file_extension);
31
32         // Validamos la extensión de la imagen
33         $valid_extension = array("png","jpeg","jpg");
34
35         if(in_array($file_extension, $valid_extension)) {
36
37             // Subimos la imagen al servidor
38             if(move_uploaded_file(
39                 $_FILES['files']['tmp_name'][$i],$carpeta)
40             ) {
41
42                 // Ejecutamos la consulta SQL
43                 $resultado->execute(array($filename,$carpeta));
44             }
45         }
46     }
47 }
48
49 <script language="javascript">window.alert("Imagen guardada exitosamente!!!
50 ");window.location="formulario.php"</script>
51 <?php
52 <body>
53 <form name="frm1" method="POST" action="" enctype="multipart/form-data">
```



#### 5. Código para visualizar archivos...

El siguiente archivo contendrá en su primera línea un fichero de conexión a MySQL y luego ejecutaremos una consulta usando PDO para obtener los registros de la tabla “*prueba* “. Además, usaremos el ciclo WHILE para devolver todos los registros que contiene dicha tabla.

Recordemos que la información devuelta de la base de datos solo será los nombres de las imágenes y tendremos que agregar la etiqueta `<img>` más la carpeta donde se subió las imágenes, en este ejercicio sería la carpeta “*subidas* “.

```
71 <?php
72 $sql2="select * from prueba";
73 $resultado1 = $base->prepare($sql2);
74 $resultado1->execute();
75
76 while ($consulta1=$resultado1->fetch(PDO::FETCH_ASSOC)) {
77 ?>
78 " width
   ='200' height='300'>
79 <?php
80 }
81 ?>
82 </body>
83 </html>
```

#### Caso de Estudio “Prueba entrevista de trabajo”.

Usted es un programador de software que recién adquirió su título profesional, ha sido llamado a una entrevista de trabajo en la empresa TECNO SOFT, donde le realizarán una prueba técnica, usted deberá programar el 100% de la solicitud según el planteamiento del problema, lea las instrucciones a continuación...

Se requiere que programe en tecnologías PHP y MYSQL que permita registrar la información de empleados de la empresa, los datos que cada empleado deberá suministrar son:

Nombre, apellido, edad, fecha de nacimiento, fotografía, sexo, título profesional, descripción de su perfil.

Usted deberá crear un formulario HTML que permita a cada empleado cargar todos estos datos.

El formulario deberá tener los elementos necesarios de tipo *text*, *number*, *file*, *textarea* necesarios para su correcto funcionamiento, una vez se llenen todos los datos solicitados y de clic en el botón “*Enviar*”, se deberá almacenar toda la información en la base de datos y mediante un *script* mostrar un mensaje que informe al empleado que su información se registró exitosamente.



INSTRUCTIVO PARA LA EJECUCIÓN DE LA FPI

← → ↻ ⓘ localhost/hv/hv.php

...:DATOS DE ASPIRANTE:...  
Nombre:   
Apellido:   
Edad:   
Fecha de nacimiento: dd/mm/aaaa   
Foto:  No se eligió ningún archivo  
Sexo:   
Profesión:   
  
Perfil:



A continuación, mediante un botón llamado “mostrar datos”, se deberá visualizar toda la información del empleado como se muestra en el siguiente mockup.

...:Datos de empleado:...  
  
Nombre: XXXXXX XXXXXX  
Apellido: XXXXXX XXXXXX  
Edad: XX  
Fecha de Nacimiento: XXX/XX/XX  
Sexo: XXXXX  
Profesión: XXXXXXXX XXXXXX  
Perfil: XXXXXXXXXXXX XXXXX XXXX

¡Manos al código!

Listado de recursos,  
materiales, equipos,  
herramientas

- Computador, Tablet o smartphone (celular).
- Conexión a internet o datos de celular.
- XAMPP correctamente instalado.
- Gestor de código fuente (el que usted prefiera).
- Actitud positiva y propositiva.

|  |   |  |
|--|---|--|
|  | <p align="center"><b>CENTRO DE GESTIÓN Y DESARROLLO SOSTENIBLE<br/>SURCOLOMBIANO</b></p> <p align="center"><b>INSTRUCTIVO PARA LA EJECUCIÓN DE LA FPI</b></p> |  |
|--|---|--|

| Tipo de Evidencia                  | Desempeño   |  | Conocimiento |  | Producto | X | No produce entrega de evidencia |  |
|------------------------------------|---|--|--------------|--|----------|---|---------------------------------|--|
| <b>Producto y Forma de Entrega</b> | <p><b>Entregable:</b><br/>Archivos fuentes que resuelvan el caso de estudio.</p> <p><b>Forma de entrega:</b><br/>Link habilitado por el instructor en plataforma TERRITORIUM.</p> |  |              |  |          |   |                                 |  |
| <b>Fecha de Inicio:</b>            | 14/02/2022  |  |              |  |          |   |                                 |  |
| <b>Fecha de Fin:</b>               | 14/02/2022  |  |              |  |          |   |                                 |  |
| <b>Intensidad /duración</b>        | 2 horas   |  |              |  |          |   |                                 |  |