

# DESARROLLO DE APLICACIONES EN INTERNET

## LABORATORIO N°05

### Introducción a PHP

**Objetivos:**

- Desarrollo de aplicaciones en PHP

**Equipos, Materiales, Programas y Recursos:**

- PC con Sistema Operativo Windows o Linux
- Laragon

**Introducción:**

En la presente sesión se detalla los fundamentos para el desarrollo de formularios en php.

**Seguridad:**

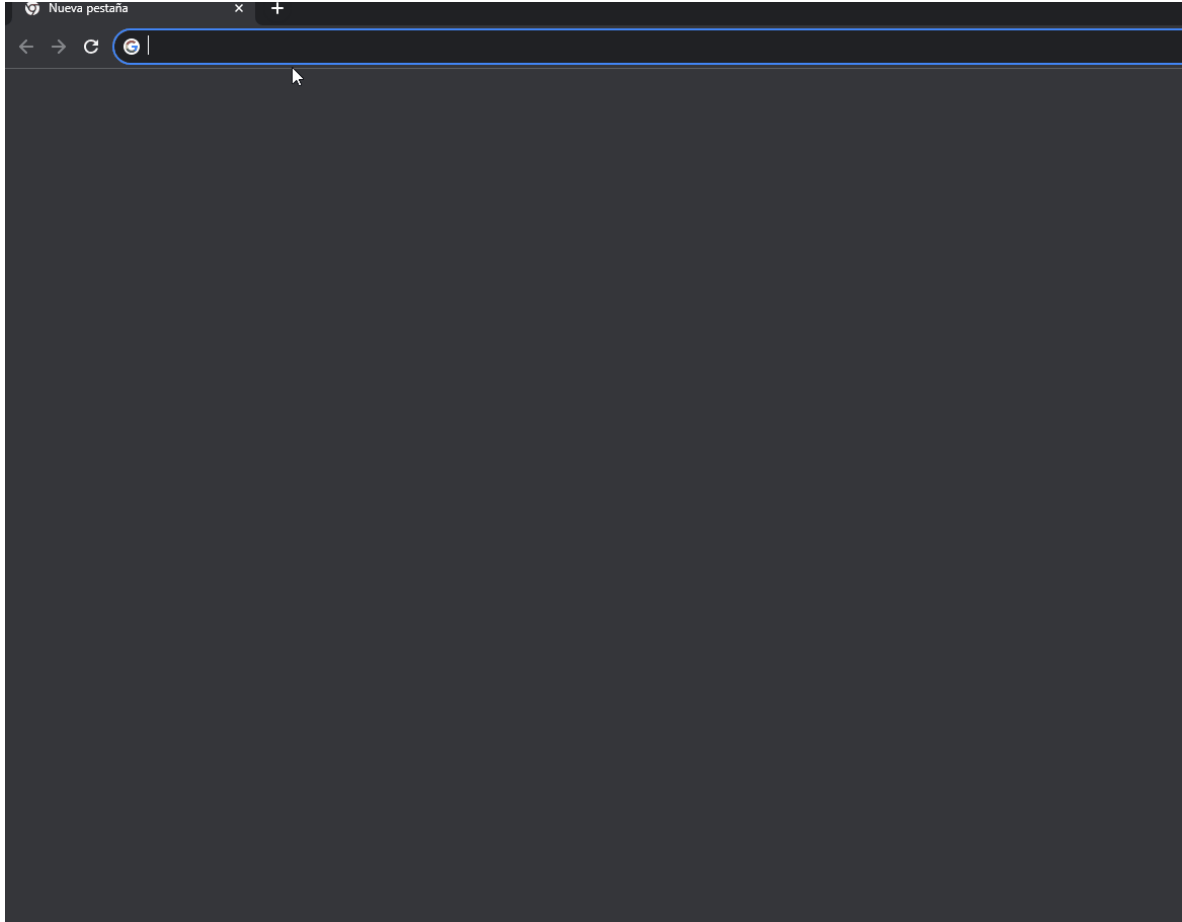
- Ubicar maletines y/o mochilas en el gabinete al final de aula de laboratorio.
- No ingresar con líquidos ni comida al aula de laboratorio.
- Al culminar la sesión de laboratorio, apagar correctamente la computadora y el monitor.

**Preparación:**

Durante el desarrollo de los temas de clase se tendrán ejercicios explicativos en cada uno de los puntos, ello le dará a la sesión una interacción de la teoría y la parte práctica, ya que en todo el momento el alumno podrá comprobar en su propia PC, todos los ítems del manual.

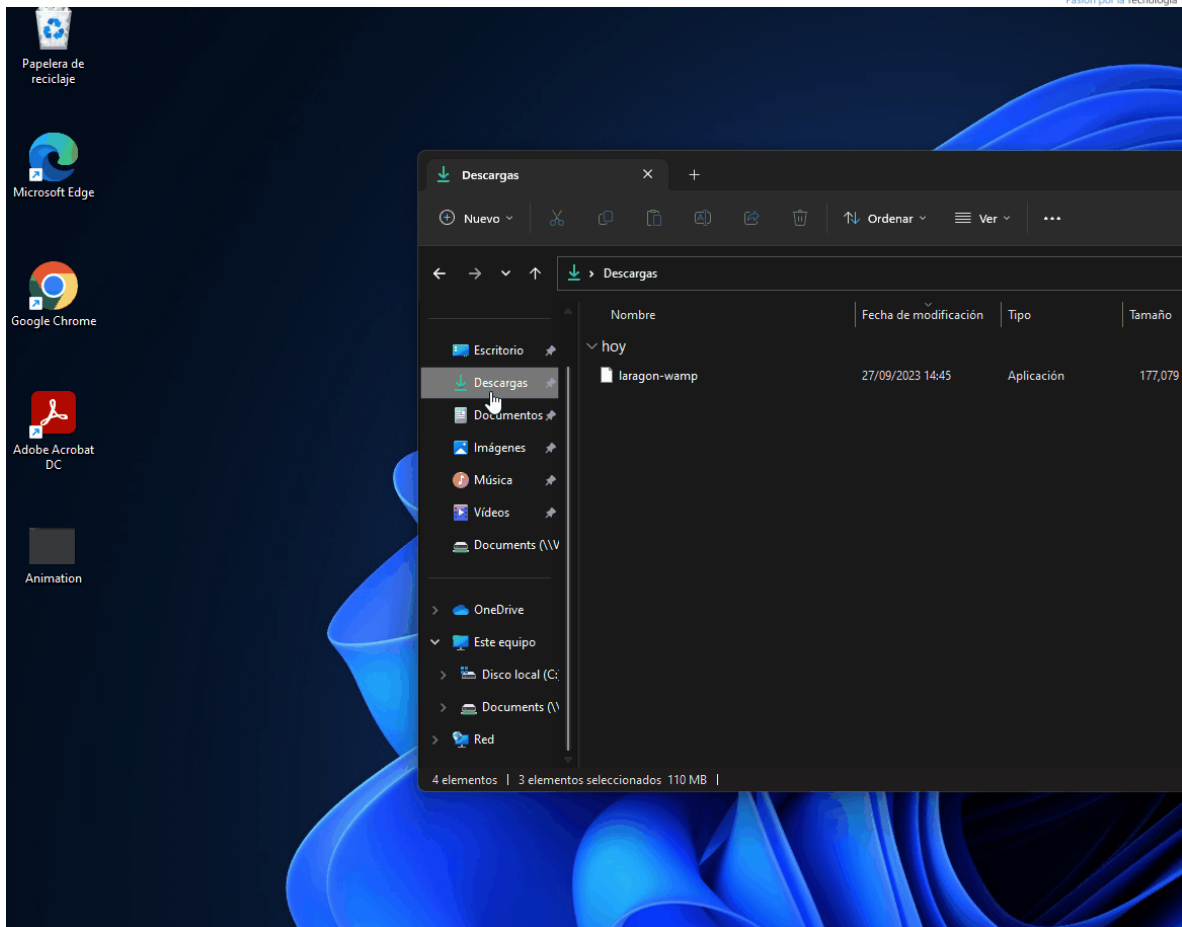
### **Procedimiento y Resultados:**

**1.- Primero instalamos el entorno de desarrollo , como ya es sabido necesitamos un servidor web para el desarrollo web en php , en nuestro caso instalaremos Laragon ,para lo cual debemos acceder a la pagina web de laragon y descargarlo.**

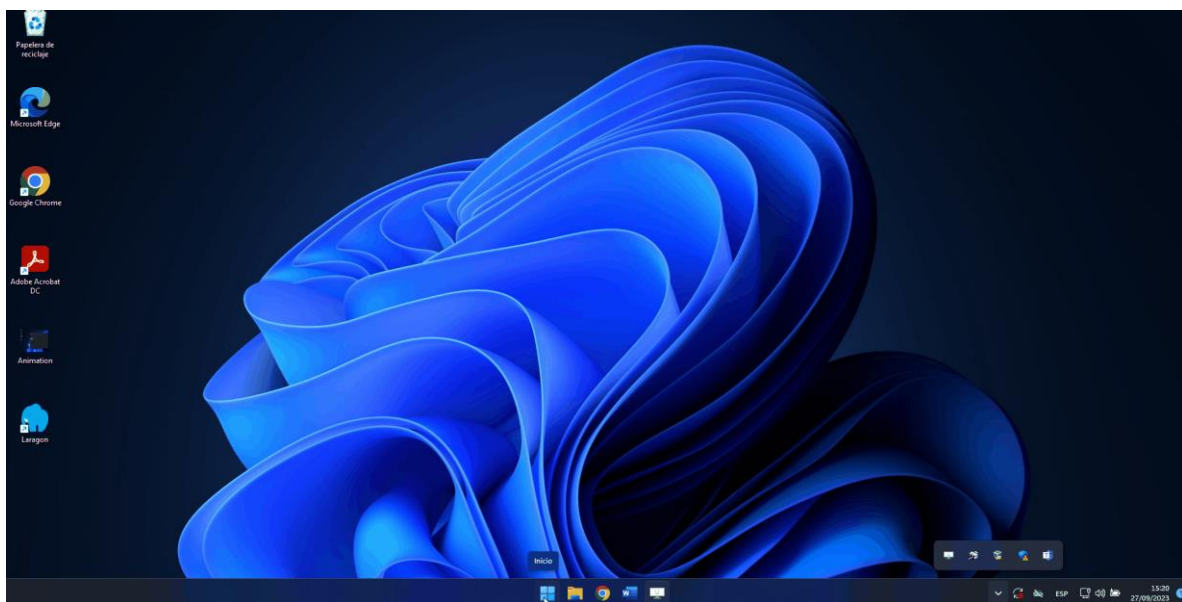


**2.- Una vez descargado debemos instalarlo con los parámetros por defecto.**

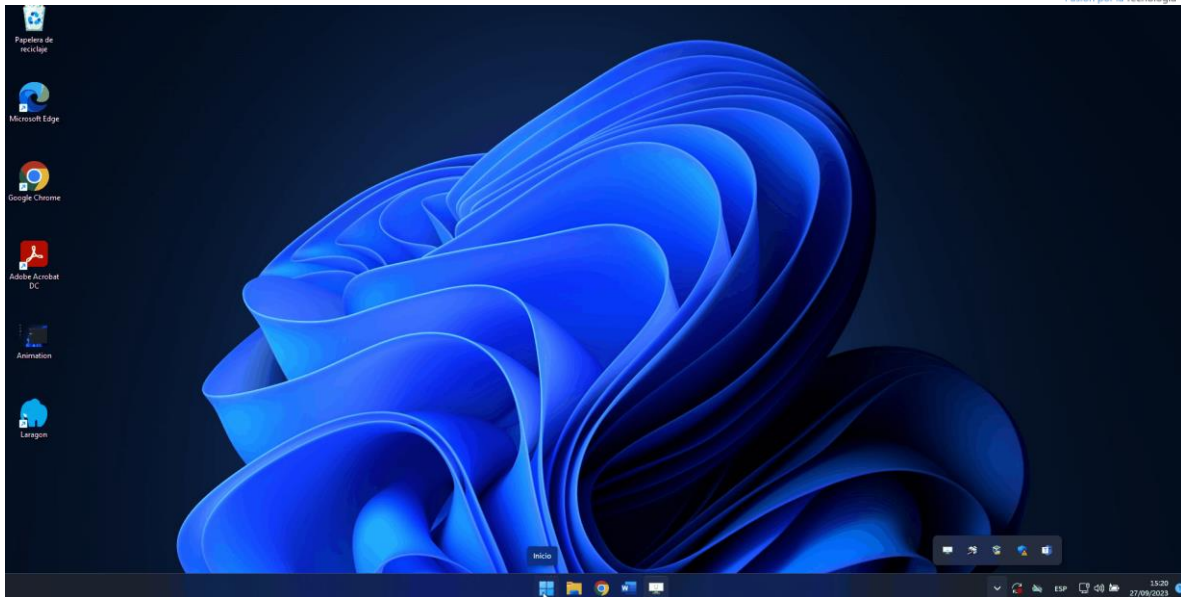




**3.- Ahora debemos iniciar laragon siguiendo los pasos de la imagen.**



**4.- Ahora creamos una carpeta con el nombre lab05 en la ruta donde instalamos laragon en nuestro caso C:/Laragon/www/.**



*Luego crearemos los siguientes ejercicios.*

## Etiqueta form

Todo formulario debe estar encerrado entre el par de etiquetas **<FORM>** y **</FORM>**, y debe ser ubicado en el cuerpo de cualquier documento *HTML*, es decir, entre el par de etiquetas **<BODY>** y **</BODY>**. Esta etiqueta **<FORM>** presenta tres atributos posibles:

Comando	Descripción
<b>ACTION</b>	el valor de este parámetro es la URL del programa o guión en el <i>Servidor Web</i> utilizado para procesar la información recolectada.
<b>METHOD</b>	puede asumir el valor GET o el valor POST, y definen la manera en la cual los datos son transferidos al servidor.
<b>ENCTYPE</b>	este atributo está reservado para que la información viaje en forma encriptada a través de Internet.

Estructura

```
<form action="mailto:direcciondelcorreo@correo.com" method="post" enctype="text/plain">  
  
</form>
```

## Method GET y POST

### Get

Cuando enviamos datos con el método GET, estos son enviados en el URL y cuando la página que solicitamos termine de cargar estos datos serán visibles para el usuario. Veamos un ejemplo de esto:

```
<a href="http://localhost/pagina.php?categoria=ropa&producto=3">ver</a>
```

### Post

Al contrario de GET, POST solo funciona enviando la información mediante formularios y no la muestra en el URL. La información que es enviada por POST va en el cuerpo de la solicitud HTTP y esta puede ir o no encriptada. Veamos un ejemplo de cómo manejar la información enviada por este método en PHP.

```
<form action="pagina.php" method="POST" >  
  Usuario: <input type="text" name="usuario" />  
  Clave: <input type="password" name="clave" />  
  <input type="submit" value="Enviar" />  
</form >
```

### METODO POST Y GET EN PHP

Cuando se envía un formulario a un script de PHP, la información de dicho formulario pasa a estar automáticamente disponible en el script. Existen algunas formas de acceder a esta información, por ejemplo:

#### Ejemplo #1 Un formulario HTML sencillo

```
<form action="foo.php" method="post">  
  Nombre usuario: <input type="text" name="username" /><br />  
  Email: <input type="text" name="email" /><br />  
  <input type="submit" name="submit" value="¡Enviarme!" />  
</form>
```

## Ejemplo #2 Acceso a datos de un formulario HTML sencillo con POST

```
<?php
echo $_POST['username'];
echo $_REQUEST['username'];
?>
```

## isset

Determina si una variable está definida o no

```
$var = '';
```

// Esto evaluará a TRUE así que el texto se imprimirá.

```
if (isset($var)) {
    echo "Esta variable está definida, así que se imprimirá";
}
```

## Actividad 01

```
ejemplo01.php x
<!DOCTYPE html>
<html>
<head>
    <title>Ejercicio 01 -Operaciones con PHP</title>
</head>
<body>
    <form method="post" action="">
        Ingresar Valor 1:<br>    <input type="text" name="valor1"><br>
        Ingresar Valor 2:<br>    <input type="text" name="valor2"><br>
        Resultado :<br>
        <input type="submit" name="Suma" value="Suma">
    </form>

    <?php
    //declaracion de variables
    $numero1=$_POST['valor1'];
    $numero2=$_POST['valor2'];
    $suma=$numero1 +$numero2;
    echo "El Resultado de la Suma es  : ".$suma;

    ?>
</body>
</html>
```

## Actividad 01

## Crear la página HTML

```
Ejemplo03.html
<!DOCTYPE html>
<html>
<head>
    <title>Estructura de formularios</title>
</head>
<body>
    <form method="post" action="http://localhost/Sesion_05/Producto.php">
        Valor 1:<br>    <input type="text" name="valor1"><br>
        Valor 2:<br>    <input type="text" name="valor2"><br>
        <br>
        <input type="submit" name="producto" value="Calcular">
        <input type="reset" name="Cancelar" value="Cancelar">
    </form>
</body>
</html>
```

## Crear la Pagina PHP

```
Producto.php
1 <?php
2     $numero1=$_POST['valor1'];
3     $numero2=$_POST['valor2'];
4     $producto= $numero1 *$numero2;
5     echo "El Producto de la Operacion es  : ".$producto;
6
7
8 ?>
```



# Analizar, Diseñar y Desarrollar:

1. Una empresa ha decidido otorgar a sus vendedores una bonificación de S/. 50 por cada hijo en edad escolar. El sueldo bruto se calcula sumando un sueldo básico de S/. 600 más una comisión que es igual al 7.5% del importe total vendido del mes y más la bonificación. Por ley, todo vendedor está sujeto a un descuento del 11% del sueldo bruto. Dados el importe total vendido del mes y la cantidad de hijos en edad escolar, diseñe un programa que determine la comisión, la bonificación, el sueldo bruto, el descuento y el sueldo neto de un vendedor.

```
1  <?php
2  // Datos del vendedor
3  $sueldoBasico = 600; // Sueldo básico
4  $porcentajeComision = 7.5; // Porcentaje de comisión
5  $bonificacionPorHijo = 50; // Bonificación por hijo en edad escolar
6  $descuentoPorLey = 11; // Porcentaje de descuento por ley
7
8  // Datos del mes (ejemplo)
9  $importeTotalVendido = 2000; // Importe total vendido del mes
10 $cantidadHijos = 2; // Cantidad de hijos en edad escolar
11
12 // Cálculos
13 $comision = ($porcentajeComision / 100) * $importeTotalVendido;
14 $bonificacion = $bonificacionPorHijo * $cantidadHijos;
15 $sueldoBruto = $sueldoBasico + $comision + $bonificacion;
16 $descuento = ($descuentoPorLey / 100) * $sueldoBruto;
17 $sueldoNeto = $sueldoBruto - $descuento;
18
19 // Mostrar resultados
20 echo "Comisión: S/. " . $comision . "<br>";
21 echo "Bonificación: S/. " . $bonificacion . "<br>";
22 echo "Sueldo Bruto: S/. " . $sueldoBruto . "<br>";
23 echo "Descuento: S/. " . $descuento . "<br>";
24 echo "Sueldo Neto: S/. " . $sueldoNeto;
25 ?>
26
```

2. Una fábrica de gaseosas ha decidido rebajar el precio de su gaseosa de 3 litros en 5% con la finalidad de igualar los precios de la competencia. Adicionalmente, ofrece un descuento especial igual al 7% del importe de la compra. Además, como incentivo especial, obsequia dos caramelos por cada gaseosa adquirida. Dados el precio actual de la gaseosa y la cantidad de unidades adquiridas, diseñe un programa que determine el nuevo precio de la gaseosa, el importe de la compra, el importe del descuento, el importe a pagar y el obsequio.

```
1  <?php
2  // Datos de la gaseosa
3  $precioActualGaseosa = 10; // Precio actual de la gaseosa de 3 litros
4
5  // Datos de la oferta
6  $descuentoPorCompetencia = 5; // Porcentaje de descuento por competencia
7  $descuentoEspecial = 7; // Porcentaje de descuento especial
8  $caramelosPorGaseosa = 2; // Cantidad de caramelos por gaseosa
9
10 // Datos de la compra (ejemplo)
11 $cantidadGaseosas = 3; // Cantidad de gaseosas adquiridas
12
13 // Cálculos
14 $nuevoPrecioGaseosa = $precioActualGaseosa * (1 - ($descuentoPorCompetencia / 100));
15 $importeCompra = $nuevoPrecioGaseosa * $cantidadGaseosas;
16 $descuento = ($descuentoEspecial / 100) * $importeCompra;
17 $importeAPagar = $importeCompra - $descuento;
18 $caramelosObsequio = $caramelosPorGaseosa * $cantidadGaseosas;
19
20 // Mostrar resultados
21 echo "Nuevo Precio de la Gaseosa: S/. " . $nuevoPrecioGaseosa . " por 3 litros<br>";
22 echo "Importe de la Compra: S/. " . $importeCompra . "<br>";
23 echo "Descuento: S/. " . $descuento . "<br>";
24 echo "Importe a Pagar: S/. " . $importeAPagar . "<br>";
25 echo "Caramelos de Obsequio: " . $caramelosObsequio;
26 ?>
27
```

[https://github.com/Zelvalla/Desarrollo\\_Web.git](https://github.com/Zelvalla/Desarrollo_Web.git)

## OBSERVACIONES

- Validación de entrada: el procedimiento especificado aún no incluye la validación de entrada del usuario (como asegurarse de que el valor ingresado sea un número válido). En aplicaciones reales, la validación de los datos de entrada es fundamental para evitar errores y ataques de seguridad.
- Escalabilidad y eficiencia: en aplicaciones y problemas de producción más grandes, la eficiencia y la escalabilidad del código se vuelven críticas. Las operaciones que involucran grandes conjuntos de datos o procesos intensivos pueden requerir estrategias avanzadas para garantizar que el programa se ejecute rápidamente y no consuma recursos del servidor.

## CONCLUSIONES

- Automatización eficiente: lenguajes de programación como PHP permiten a las empresas automatizar eficientemente procesos complejos como el cálculo de salarios, descuentos y bonificaciones. Esto aumenta la precisión y reduce el error humano en los cálculos financieros.
- Adaptabilidad a diferentes escenarios: los programas PHP pueden adaptarse fácilmente a diferentes situaciones y escenarios. En los ejemplos dados, puede ver cómo se pueden utilizar los mismos principios de programación para resolver problemas muy diferentes, desde cálculos de nómina hasta estrategias de marketing de ventas de productos.