

# CURSO .PHP

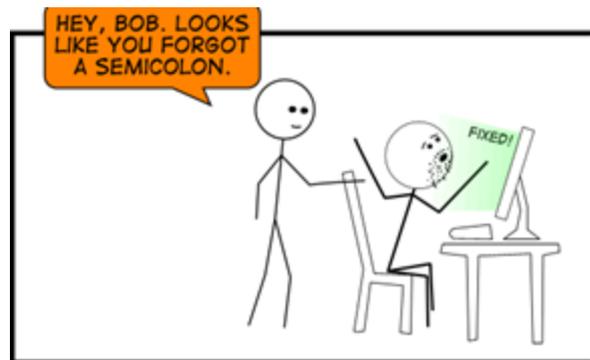
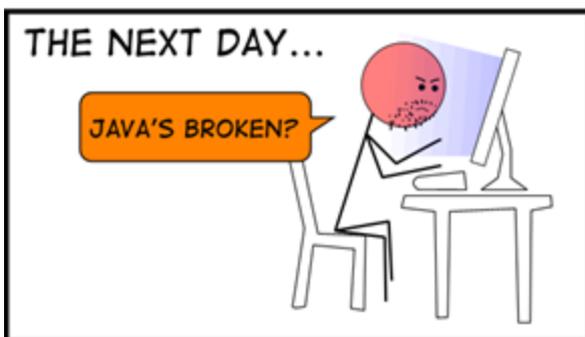
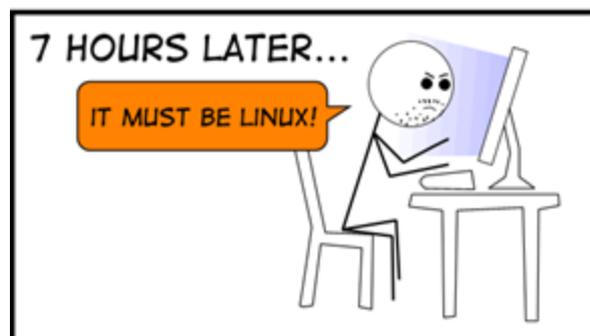
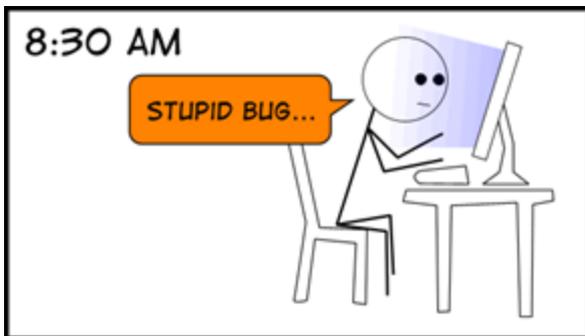
## ENVÍO DE INFORMACIÓN AL SERVIDOR



Autor: Jon Vadillo  
Modificado por: Inés  
Larrañaga

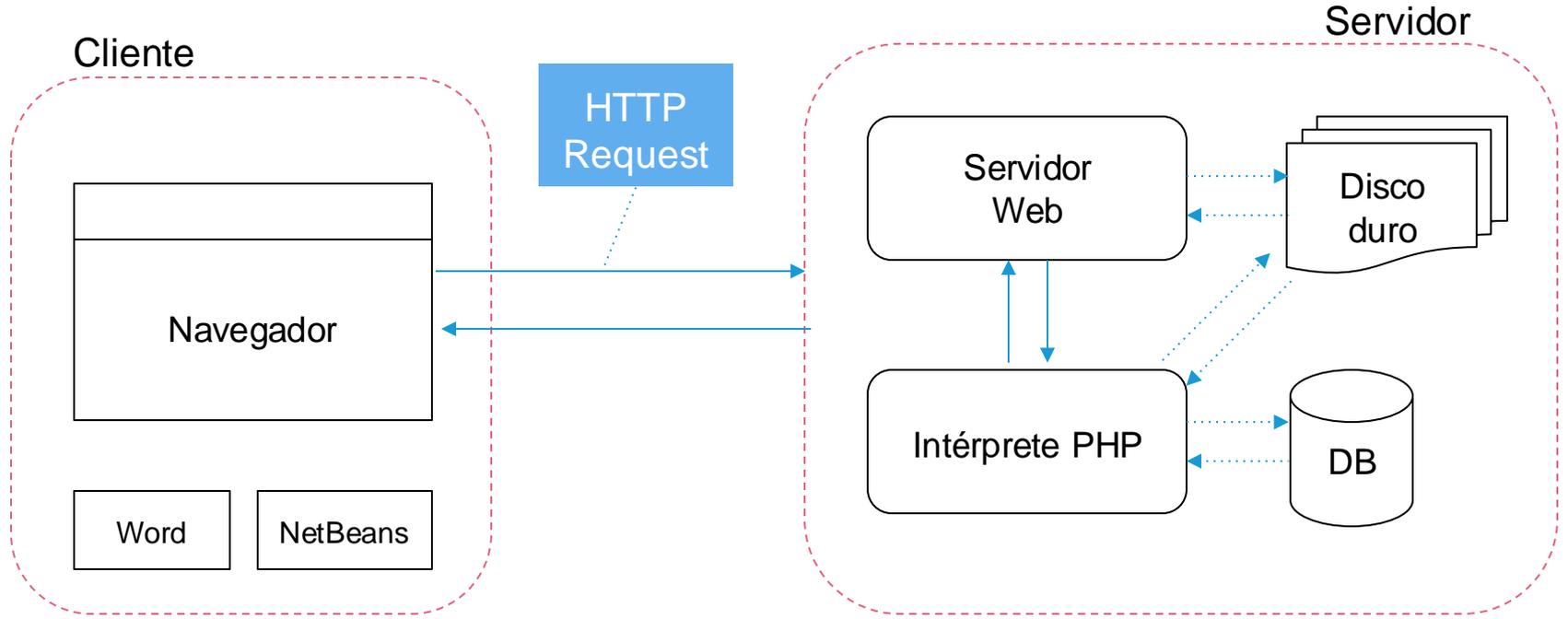
# Contenidos

- Fundamentos básicos
- GET
- POST
- Enlaces HTML



Un **formulario web** permite al usuario introducir datos los cuales son **enviados a un servidor** para ser procesados.

# Fundamentos básicos



# Fundamentos básicos

- Existen dos métodos HTTP para enviar información al servidor:
  - GET
  - POST
- PHP proporciona tres **variables superglobales** para acceder a la información:
  - `$_REQUEST`
  - `$_GET`
  - `$_POST`

# HTTP GET

- Envía la información en la URL.
- La información se codifica en un esquema de **clave - valor** separados por un ampersand & que comienza a partir del interrogante:

`http://miapp.com/index.php?key1=value1&key2=value2`

# HTTP GET

- La información enviada **es visible**. No se debe enviar información sensible.
- No se pueden enviar datos binarios (imágenes u otros archivos).
- Los datos se manejan mediante el array asociativo `$_GET`

`http://miapp.com?nombre=Julen` →

`$_GET["nombre"]`

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<form action="formulario.php" method="get">
  Nombre: <input type="text" name="nombre"><br>
  Email: <input type="text" name="email"><br>
  <input type="submit" value="Enviar">
</form>
Hola <?php echo $_GET["nombre"] ?><br>
Tu email es: <?php echo $_GET["email"] ?>
</body>
</html>
```

formulario.php?nombre=lorea&email=lore%40gmail.com

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<form action="formulario.php" method="get">
  Nombre: <input type="text" name="nombre"><br>
  Email: <input type="text" name="email"><br>
  <input type="submit" value="Enviar">
</form>
Hola <?php echo $_GET["nombre"] ?><br>
Tu email es: <?php echo $_GET["email"] ?>
</body>
</html>
```

## Controlar la existencia

- Es importante controlar mediante la función `isset()` que las variables a las que se intenta acceder existen:

```
<?php

if(isset($_GET["nombre"])) {
    $nombre = $_GET["nombre"]
}

?>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<form action="formulario.php" method="get">
  Nombre: <input type="text" name="nombre"><br>
  Email: <input type="text" name="email"><br>
  <input type="submit" value="Enviar">
</form>
Hola <?php isset($_GET["nombre"]) ? print $_GET["nombre"] : ""; ?><br>
Tu email es: <?php isset($_GET["email"]) ? print $_GET["email"] : ""; ?>
</body>
</html>
```

formulario.php?nombre=lorea&email=lore%40gmail.com

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<form action="formulario.php" method="get">
  Nombre: <input type="text" name="nombre"><br>
  Email: <input type="text" name="email"><br>
  <input type="submit" value="Enviar">
</form>
Hola <?php isset($_GET["nombre"]) ? print $_GET["nombre"] : ""; ?><br>
Tu email es: <?php isset($_GET["email"]) ? print $_GET["email"] : ""; ?>
</body>
</html>
```

# HTTP POST

- Envía la información en el cuerpo de la petición HTTP Request, no en la cabecera.
- La información enviada no es visible. Es el método recomendado para enviar información sensible.
- Puede enviarse texto plano y datos binarios (imágenes, archivos,...)
- Los datos se manejan mediante el array asociativo `$_POST`.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<form action="formulario.php" method="post">
  Nombre: <input type="text" name="nombre"><br>
  Email: <input type="text" name="email"><br>
  <input type="submit" value="Enviar">
</form>
Hola <?php isset($_POST["nombre"]) ? print $_POST["nombre"] : ""; ?><br>
Tu email es: <?php isset($_POST["email"]) ? print $_POST["email"] : ""; ?>
</body>
</html>
```

formulario.php

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<form action="formulario.php" method="post">
  Nombre: <input type="text" name="nombre"><br>
  Email: <input type="text" name="email"><br>
  <input type="submit" value="Enviar">
</form>
Hola <?php isset($_POST["nombre"]) ? print $_POST["nombre"] : ""; ?><br>
Tu email es: <?php isset($_POST["email"]) ? print $_POST["email"] : ""; ?>
</body>
</html>
```

formulario.php

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<form action="formulario.php" method="get">
  Nombre: <input type="text" name="nombre"><br>
  Email: <input type="text" name="email"><br>
  <input type="submit" value="Enviar">
</form>
Hola <?php isset($_GET["nombre"]) ? print $_GET["nombre"] : ""; ?><br>
Tu email es: <?php isset($_GET["email"]) ? print $_GET["email"] : ""; ?>
</body>
</html>
```

# Enlaces HTML

- Podemos construir enlaces que contengan información y acceder mediante GET.

```
<a href="formulario.php?nombre=mikel&edad=44">Enlace</a>
```

```
<?php
    $nombre = $_GET["nombre"];
    $edad = $_GET["edad"];
    echo $nombre . "<br>";
    echo $edad . "<br>";
?>
```

# Hands on!

01. Crea un convertidor de grados Celsius a Farenheit (y viceversa). La aplicación pedirá una cantidad al usuario y la unidad mediante un formulario. Una vez enviado, mostrará el resultado de la conversión realizada. El formulario siempre se mostrará, de forma que el usuario pueda seguir introduciendo valores.

**Resultado de la conversión (44 celsius): 111.2**

Introduce la temperatura:

Indica la unidad de la temperatura introducida:

Enviar

# Hands on!

02. Crea una aplicación que simule una calculadora. La calculadora estará representada por un formulario con dos campos numéricos y un desplegable para indicar la operación que se desea realizar (suma, resta, multiplicación o división). La aplicación deberá mostrar el resultado de la operación realizada y mostrar un mensaje de error cuando se intente dividir entre cero.

## Calculadora

Último resultado: 10

Primer número:

Segundo número:

Seleccione la operación deseada:

# Hands on!

03. Crea un array con 20 números aleatorios entre el 1 y el 10 utilizando la función `random_int()`. El usuario introducirá en el formulario un número y al darle a “Enviar” la aplicación comprobará cuántas veces aparece el número introducido en el array.

El número aparece 5 veces en el array.

Introduce tu número:

# Hands on!

04. Crea una pantalla de login con dos campos: usuario y contraseña. El programa deberá validar los datos de acceso contra un diccionario de datos como el de la imagen. El programa deberá mostrar un mensaje de bienvenida con su nombre y apellidos en caso de que los datos sean correctos.

También indicará si el usuario introducido no existe o si la contraseña introducida es incorrecta.

```
$usuarios = array(  
    "user1" => array (  
        "nombre" => 'Ane',  
        "apellidos" => 'López',  
        "password" => '123Abc'  
    ),  
  
    "user2" => array (  
        "nombre" => 'Amaia',  
        "apellidos" => 'Otsoa',  
        "password" => '456Xyz'  
    )  
);
```

# Hands on!

05. Desarrolla una tienda online con un catálogo de productos y un carrito de la compra. Cada producto tendrá la siguiente información: nombre, descripción y precio (puedes almacenar otra información extra que te sirva de ayuda, como un ID). Se podrán adquirir tantas unidades de cada producto como se desee (también podrá eliminarse un producto añadido o una unidad del mismo).

La aplicación mostrará un formulario en el que indicar la cantidad de cada producto. Al hacer click en “Enviar” la aplicación le llevará al usuario a otra página donde únicamente mostrará el valor total de la compra realizada y un enlace para volver a la página anterior.

# Hands on!

## Catálogo de productos

Nombre	Descripción	Precio	Cantidad
Logitech K120	Teclado multimedia USB plug&play, trackPoint Caps (10pk, Soft Dome)	25.99	<input type="text" value="0"/>
Lenovo LI5	El mouse Lenovo 300 compacto inalámbrico es el accesorio perfecto para cualquier persona que desee un mayor control y libertad	12.99	<input type="text" value="0"/>
Monitor LG X10	LCD con retroiluminación LED ThinkVision T1714p Square de 17 pulgadas	179.99	<input type="text" value="0"/>
Monitor Lenovo Q24i	Pantalla de 60,45 cm (23,8") Funciones como AMD FreeSync	172	<input type="text" value="0"/>
ThinkPad X1 Extreme	ThinkPad X1 Extreme de 2.ª generación gestiona exigentes tareas informáticas sin problemas. Con pantalla táctil 4K OLED	1200	<input type="text" value="0"/>

Comprar

## Precio total:

El importe total de la compra realizado es de 2127.95 euros.

## Detalle de la compra

- Logitech K120 (1)
- Lenovo LI5 (2)
- Monitor LG X10 (2)
- Monitor Lenovo Q24i (3)
- ThinkPad X1 Extreme (1)

Volver a la tienda

# Sources

- The PHP Group: <https://www.php.net/>